10.0 FEDERAL CONVENTIONS FOR USING ASC X12 TRANSACTION SETS

This part of the Federal Implementation Guidelines defines the Federal transaction set conventions. It includes the instructions for implementing the control and security structures and definitions of the usage indicators and applicable codes.

This version of Part 10 of the Federal Implementation Guidelines, incorporates the requirements and features of ANSI ASC X12 Version 004 Release 010 standards. It supersedes and cancels the 28 August 1997 version of Part 10.

Except where specifically otherwise indicated, this document directs how the agencies, components and activities of the United States Federal government will exchange Electronic Data Interchange (EDI) data formatted in accordance with the provisions of the ANSI ASC X12 standards.

10.1 INTRODUCTION

The power of the American National Standards Institute (ANSI) Accredited Standards Committee (ASC) X12 standard is in its building block concept, which standardizes the essential elements of business transactions. The concept is analogous to a “standard bill of materiel and the construction specifications,” which gives the architect flexibility in what can be designed with standardized materiel and procedures. The Electronic Data Interchange (EDI) system designer, like the architect, uses the ASC X12 standards to build business transactions that are often different because of their function and yet utilize the ASC X12 standards. The “bill of materiel and the construction specification” of ASC X12 are the standards found in the published technical documentation.

ASC X12.3, ***December 1997***—The *Data Element Dictionary* specifies the data elements used in the construction of the segments that comprise the transaction sets developed by ASC X12.

ASC X12.5, ***December 1997***—The *Interchange Control Structure* provides the interchange control segment (also called an envelope), consisting of a header and trailer, for the EDI transmission; it also provides a structure to acknowledge the receipt and processing of the envelope.

ASC X12.6, ***December 1997***—The *Application Control Structure* defines the basic control structures, syntax rules, and semantics of EDI.

ASC X12.22, ***December 1997***—The *Data Segment Directory* provides the definitions and specifications of the segments used in the construction of transaction sets developed by ASC X12.

ASC X12.58, ***December 1997***—The *Security Structures* define the data formats for authentication, encryption, and assurances in order to provide integrity, confidentiality, verification and non-repudiation of origin for two levels of exchange of Electronic Data Interchange (EDI) formatted data: functional group and transaction set level.

X12.59, ***December 1997***—The *Implementation of EDI Structure/Semantic Impact* provides a clear distinction between the syntax of X12 structures and the semantics of transaction set usage.

X12C/TG1/95-65—*Technical Report Reference Model for the Acknowledgment and Tracking of EDI Interchanges* summarizes the use of the ANSI ASC X12 control elements and standards for the acknowledgment and tracking of EDI interchanges.

International Telecommunication Union-Telecommunication Standardization Sector (ITU-T) Recommendation X.509 *(1993)/ ISO/IEC 9594-8 (1995), Information Technology-Open Systems Interconnection-The Directory: Authentication Framework. The Directory*, defines a framework for the provision of authentication services by the Directory to its users. It specifies the form of authentication information held by the Directory, describes how authentication information may be obtained from the Directory, states the assumptions made about how authentication information is formed and placed in the Directory, defines three ways in which applications may use authentication information to perform authentication, and describes how other security services may be supported by authentication.

In addition to using existing standards to build specific transactions, the standards may be used to provide control and tracking of interchanges if accomplished in a specific standardized approach. ANSI ASC X12 has defined and approved several control structures and Transaction Sets intended to augment EDI auditing and control systems. It is the intent of these standards to provide a tracking mechanism for EDI data as it moves through the transmission cycle. Through the implementation of these tracking tools and analysis of the resulting information, delay or failures in delivery can be identified and corrected.

The work accomplished by ANSI ASC X12C in this area produced a generic acknowledgment model that has been adapted to support Federal Government EDI processes. Implementation of the acknowledgment mechanisms identified by this model will provide a basic capability to track interchanges as they flow from senders through Service Request Handlers (SRH) to receivers across the Electronic Commerce (EC)/EDI Infrastructure. (An SRH is a service provider whose primary function is to provide communications services between other components in the model.) This basic capability will provide functionality for each component to determine translation and transmission status, including current location and disposition of an interchange. Use of the implemented acknowledgment mechanisms to determine singular event status can provide components with the information necessary to obtain some level of confidence that interchanges are flowing through the infrastructure properly. Taken as a sequence of acknowledgment events, the model provides senders with a means to track interchanges from generation to delivery to a Service Request Handler at the boundary of the infrastructure, without imposing the processing and communications overhead that would be required for true application to application acknowledgments.

In addition, the implemented acknowledgment mechanisms of this model will allow individual components to build upon or enhance their internal audit trail processes.

This part of the Federal Implementation Guidelines is meant to be an overarching architecture of the control and security structure which the government is implementing in the Electronic Commerce Infrastructure (ECI) and other government EC activities. However, not all the parts of the architecture will be implemented immediately. The specifics of which parts are actually implemented will be defined in agreements between actual components in the trading network and architecture, such as Value Added Networks (VANs) and government users of the ECI.

It is not the intent of this guideline to specify how the implemented acknowledgment mechanisms are to be used. While support of these mechanisms is required, their usage between infrastructure components will be as agreed between those components. The use of certain acknowledgement mechanisms between the government and VANs and gateways is specified in an Agreement between the parties. When there is a conflict between the implementation guidance provided in the Part 10 and the Agreement, the Agreement shall take precedence. Also, the use of acknowledgments between Government Points of Translation (GPoT) and other infrastructure components can be as mutually agreed upon.

The Service Level Agreement (SLA) between the ECI and the respective government Automated Information Systems (AIS) acts in a similar manner as the Agreement between the government and VANs and gateways. Where there is a conflict between the implementation guidance provided in Part 10 and the SLA, the SLA shall take precedence.

By focusing on basic acknowledgment functionality that is independent of communications protocols, enhanced tracking of interchanges is accomplished without requiring individual components to adhere to or support a full accountability system.

For further clarification of acronyms, abbreviations, and codes, refer to ASC X12 published technical documentation. For copies, contact either the EDI focal point within your service or agency, or, alternatively, contact the administering body (see Section 1.3 of these guidelines).

10.1.1 Year 2000 Compliant Date Formatting

Data elements reflecting dates in ANSI ASC X12 version/release 004010 and beyond are capable of carrying Year 2000 (Y2K) compliant dates in the standard date format (CCYYMMDD). While the use of 004010 and higher versions are the preferred alternatives for becoming Year 2000 compliant, a methodology for distinguishing between the 20th and 21st centuries is necessary. For the purpose of Y2K compliance, the following are work-around and permanent solutions for meeting Y2K compliance.

10.1.1.1 Bridging (49/50 rule) is applicable for versions 2003-3070.

Bridging is a technique in which a year of 00 – 49 is considered to reference the year 2000, and a year of 50 – 99 is considered to reference the 1900 year. This solution requires that the AIS make modifications to decipher the bridging schemes. If necessary, those systems that employ a "bridging" scheme will need to identify ICs that are impacted and provide additional implementation notes on the usage of the bridging scheme.

Note: **The bridging solution is the only option available for systems based on versions 002003–003010**; these versions of the ANSI X12 Standard do not support the century date format.

Examples of bridging using the 6-digit date format to depict the century. [**YY**MMDD]

1. 19[**501206**] = 1950 December 06
   YYMMDD

2. 19[**591206**] = 1959 December 06
   YYMMDD

3. 20[**001206**] = 2000 December 06
   YYMMDD

4. 20[**491206**] = 2049 December 06
   YYMMDD

10.1.1.2 Date & Century Data Elements is applicable for versions 003020–003070.

Use data element 373 [Date YYMMDD] in conjunction with data element 624 [Century CC].

Use data element 1250 [Date Qualifier CC] in conjunction with data element 1251 [Date Time Period].

Note: Systems based on versions 003020–003070 can either use the bridging solution or composite date solutions (DATA ELEMENT 373 and DATA ELEMENT 624) or (DATA ELEMENT 1250 and 1251) for Y2K compliance. If the ICs impacted do not support the composite date solutions (i.e. DATA

ELEMENT 373 or 1250 marked "Not Used"), then the bridging solution is appropriate.

10.2 CONTROL SEGMENTS

In addition to communications control, the EDI interchange structure provides the standards user with multiple levels of control to ensure data integrity. It does so by using header and trailer control segments designed to identify uniquely the start and end of the interchange functional groups and transaction sets. The relationship of these control segments is shown in Figure 10.2-1. Control Segment specifications are defined in Section 10.6.

10.2.1 Description of Use

The interchange header and trailer segments (ISA/IEA) along with the optional interchange acknowledgment segments (TA1 and TA3) constitute the interchange control structure (i.e., an interchange envelope). Interchange control segments perform the following functions:

- Define data element separators, subelement separators and data segment terminators
- Provide control information
- Identify interchange sender and receiver
- Allow for authorization and security information.

The actual interchange control structure includes neither the group control structures nor the transaction control structures; these are defined by ASC X12 as application control structures, and their version and release may differ from those for the interchange envelope. An interchange envelope encompasses one or more functional groups (GS/GE), which, in turn, enclose one or more related transaction sets (ST/SE). The relationship for these structures is illustrated in Figure 10.2-1.

The purpose for GS/GE functional grouping is to provide an additional control envelope surrounding like transaction sets conforming with a unique Implementation Convention (IC). Their usage is prescribed as interchange control segments in order to present a consistent methodology for EDI within the government community and for commercial entities that conduct EDI business with the government.

Implementation Note: *The Federal Government Electronic Commerce Infrastructure (ECI) shall send and receive textual data ASCII encoded. If unencrypted binary segments are filtered, Base 64 filtering shall be used.*

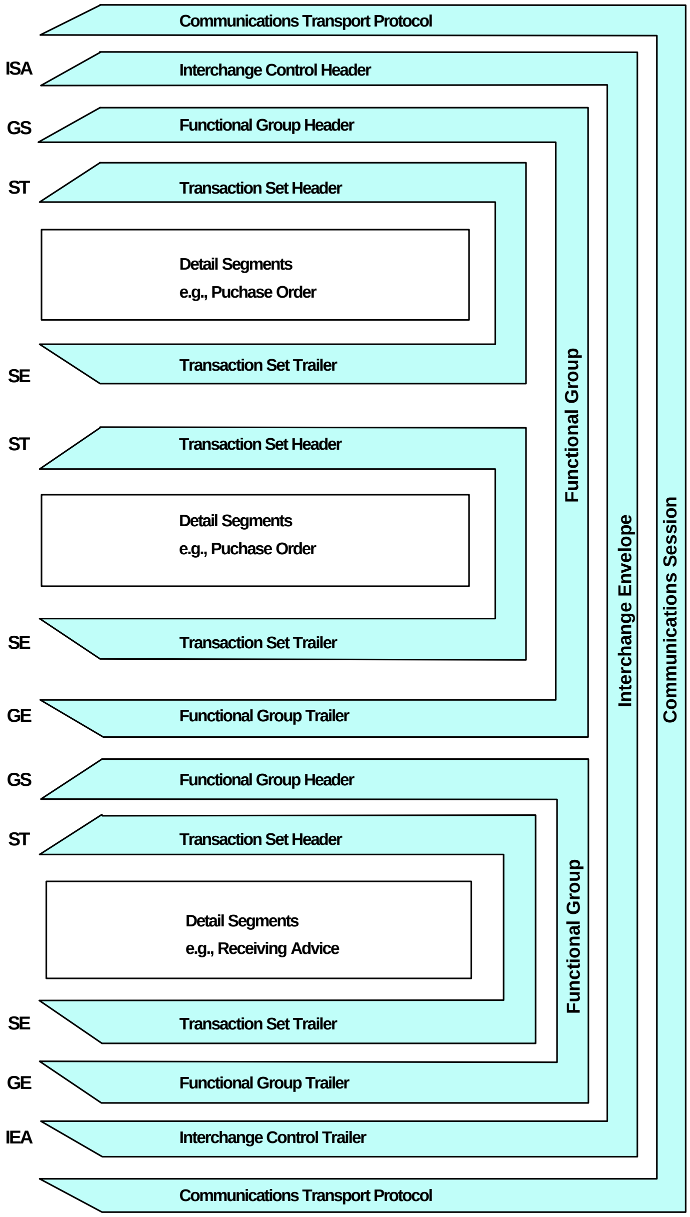


Figure 10.2-1. Hierarchical Structure

Note: When an Interchange contains TA3s, it shall contain only TA3s. The TA3s replace all Functional Groups, Security Envelopes, Transaction Headers and Trailers, as well as Detail Segments in the above diagram.

10.2.1.1 Data Element, Data Segment, and Component Data Element Separation

In ASC X12 documentation, the data element separator is graphically displayed as an asterisk (*). The actual data element separator employed within the interchange envelope dictates the value for the entire interchange. The first occurrence of the data element separator is at the fourth byte of the interchange control header. The value appearing there dictates the data element separator used through the next interchange trailer.

In a similar manner, the interchange control header establishes the value to be used for segment termination within an interchange. ASC X12 documentation represents this graphically by a new line (N/L). The first instance of segment termination occurs immediately following the ISA16 data element, and the data value occurring there sets the value for the interchange.

Designation of a component data element separator differs from the other separators in that the ISA segment provides a discrete element (ISA16) for defining the component data element separator data value.

Implementation Note:

1. ASCII hexadecimal character 1C shall be used as the segment terminator in Federal Government interchanges.

2. ASCII hexadecimal character 1D shall be used as the data element separator in Federal Government interchanges.

3. ASCII hexadecimal character 1F shall be used as the component element separator in Federal Government interchanges.

4. These characters are reserved for these purposes and shall not be used in data elements, except that they may be used in data element 785, Binary Data.

5. Failure to comply with the 1C, 1D, and 1F terminator/separator guidance will result in transaction errors.

6. All senders of transactions across the Government's EC infrastructure are required to follow this guidance.

*7. Due to reliability, throughput, and speed-of-service concerns, government and commercial trading partners are encouraged to use File Transfer Protocol (FTP) or Simple Mail Transfer Protocol (SMTP) instead of dial-up protocols to access the electronic commerce infrastructure****. These delimiters may not work with bisync communications.***

10.2.1.2 Identification of Implementation Convention

The Federal Government develops ICs based on ASC X12 standards. All entities conducting EDI business within the Government or externally with the Government shall comply with all applicable ICs. Implementation Conventions are available from National Institute of Standards and Technology acting as the secretariat for the Federal EDI Standards Management Coordinating Committee (FESMCC). Conventions on the use of interchange control structures are provided herein to document a consistent approach to control structure content. The functional group control structures include the ability to identify specific ICs to which the Transaction Sets contained within that group conform.

Interchange senders will provide the ASC X12 Version/ Release/Subrelease and IC identifier in GS08. This identifier uniquely identifies the convention to which the transaction set conforms.

Implementation Note: *Envelope control segments have few options and, except for minor tailoring, are identical for every EDI interchange. The tailoring involves the code values selected for the GS01 and GS08 elements. GS01 classifies the particular transaction set(s) within a functional group and GS08 identifies the specific IC with which the transactions contained within the group comply. (Note: The version and release identified in ISA12 pertains to the interchange control envelope, not to the contained transaction sets.)*

The Version/Release/Industry Identifier Code (GS08) is structured as follows:

Positions 1 through 6:	ANSI ASC X12 Version and Release number (e.g. 004010) upon which the IC is based.

Position 7	Organizational Scope
	A = APADE
	F = Federal
	D = DOD
	G = Government (Internal)
	R = Rework
	1 = MADES I
	2 = MADES II
	5 = SACONS 2.5

6 = SACONS 2.6

Positions 8 through 10 — Transaction Set Identifier Code (e.g. 850).

Position 11: — Derivative: A character used to define the functionality of a specific implementation of a transaction set.

If the convention is not a derivative, an underscore (_) will appear in this position.

The specific value is assigned by the functional working group responsible for the development of the IC.

Position 12: — A sequential number starting with 0 and incremented by 1 each time the convention is reissued.

An example of the Version/Release/ Industry Identifier Code for X12 Version 004010, Federal Specific IC, revision 1, Commercial Invoice (810C) is 004010F810C1.

10.2.1.3 Control Numbers

ASC X12 standards provide for syntax control on three levels: interchange, group, and transaction. Within each level, control numbers exhibit a positive match between the header segment and its corresponding trailer (i.e., ISA/IEA, GS/GE, and ST/SE). Assignment of these control numbers, at each level, is as follows:

Implementation Note: *ISA/IEA Interchange Control Numbers (ISA13/IEA02).*

1. The nine-digit interchange control number is usually assigned by the originator's translation software. Originating organizations may use any numbering scheme consistent with their business practices.

2. The scheme must provide sufficient uniqueness to identify each interchange. Unique identification is defined as the triplet: Interchange Sender ID, (ISA05, ISA06), the Interchange Receiver ID, (ISA07, ISA08) and the nine-digit Interchange Control Number (ISA13). This triplet shall be unique within a reasonably extended time frame.

3. If there is no TA3, Interchange Delivery Notice, after 2 hours, then retransmit with the same interchange control number (ISA13).

4. If an interchange is rejected, the corrected interchange shall have a new interchange control number (ISA13).

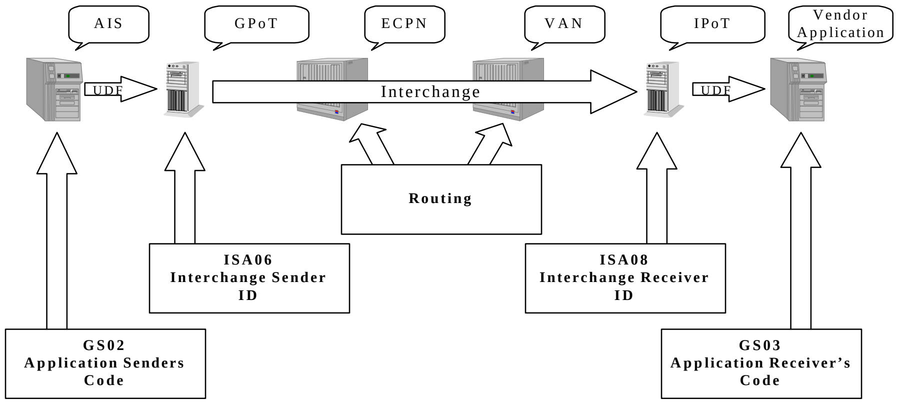
Implementation Note: *GS/GE Data Interchange Control Numbers (GS06/GE02).*

1. This is a one to nine-digit number usually assigned by the originator's translation software. This number uniquely identifies functional groups transmitted between sending and receiving application pairs. Originating organizations may use any numbering scheme consistent with their business practices.

2. The scheme must provide sufficient uniqueness to identify each functional group. The Group Control Number value (GS06), together with the Application Sender's Code (GS02), Application Receiver's Code (GS03), and Functional Identifier Code (GS01), shall be unique within an extended time frame—such as a year.

Implementation Note: *ST/SE Transaction Set Control Numbers (ST02/SE02). The originator's translation software usually assigns the transaction set control number. Originating organizations may use any numbering scheme consistent with their business practices. The scheme must provide sufficient uniqueness to identify each transaction set, within the context of the functional group.*

The control numbers within corresponding header and trailer segments must match. This provides a means to detect loss of data.



GPoT = Government Point of Translation
IPoT = Industry Point of Translation

Figure 10.3-1 Addressing Model

10.3 ADDRESSING

The purpose of addressing is to provide an unambiguous reference to a transmission's sender and intended receiver. The addressing model used by the Federal Government for ASC X12 EDI transmissions is graphically depicted in Figure 10.3-1. In this model, there is addressing for two types of transmissions. The first is an interchange. It consists of control segments and application data. The second type is application data. Application data flow from the sending to the receiving applications and is transported within an interchange. Since interchanges are assembled by the sending translation point and disassembled by the receiving translation point, the flow of an interchange is defined to be from translation point to translation point. Application data must be provided to the sending translation point by the sending application and is depicted as a User Defined File (UDF). It must also be provided to the receiving application by the receiving translation point and is also depicted as a UDF.

While the model depicts data flow from the government to a vendor, it is equally applicable in the reverse flow.

10.3.1. Interchanges

Interchanges flow between translation locations. The GPoT can be implemented as part of the government AIS, as part of the Electronic Commerce Processing Node (ECPN), or as a stand-alone function. Likewise, the Industry Point of Translation (IPoT) on the vendor side can be in the Vendor Application, as part of the VAN's services, or as a stand-alone function.

The GPoT and IPoT are addressed by the Interchange Sender ID (ISA05 and ISA06) and Interchange Receiver ID (ISA07 and ISA08) data elements. These, combined with the Interchange Control Number (ISA13), create a triplet that defines a globally unique identifier for the interchange. The ASC X12 Interchange flows between these translation points.

Implementation Note:

1. When an interchange contains one-to-one transactions, the Interchange Sender ID (ISA06) and Interchange Receiver ID (ISA08) data elements shall be the addresses of the interchange translation points (both government and non-government).

2. Translation Points (ISA06 and ISA08) shall be identified via a unique identifier from one of the sources listed as allowable codes in the ISA05 definition in section 10.6. The Data Universal Numbering System (DUNS) number and DUNS+4 are the preferred identifiers.

3. All commercial and government entities conducting business electronically shall provide their translation point (ISA06/ISA08) codes during registration.

4. In the ECI, when an interchange contains public transactions the ISA08 will be addressed individually to all certified VANs, not necessarily each IPoT. The ISA06 will contain the ECPN's address.

10.3.2 Application Sender and Receiver Codes

Application data is transported within the interchange via groups. Group addressing (GS02/GS03) must define the user application end points shown in Figure 10.3-1 as the AIS and the Vendor

Application. These addresses are locally unique and are defined between the translation point and its customers. The data that flows between the translation points and the Application Senders and Receivers are not defined by ASC X12, but are in a format agreed between the applications and their translation points.

ASC X12 standards provide for the identification of senders and receivers on two levels, the interchange and the group. The group level identifies application senders and receivers. Depending on where translation is performed, the sender/receiver IDs may be the same at the interchange and group levels and may use any number of available naming schemes.

At the GS/GE level, DUNS and DUNS+4 are recommended, especially for identifying government organizations. Other identifiers may be used.

A DUNS number may be acquired from Dun and Bradstreet and the plus 4 portion of the number is assigned and maintained internally by each entity. Specific use of these numbers is provided for in the control structures section of this document.

Implementation Note:

1. The GS02/03 identifiers need be unique only within the context of the associated ISA address.

2. All commercial and government entities conducting business electronically shall provide their Application Sender and Receiver (GS02/GS03) codes during registration.

10.4 ACKNOWLEDGMENTS

The successful conduct of business via EDI requires that trading partners be able to determine when transactions were received, not received, received in error, or otherwise did not complete the transmission or receiver application processing cycle. The generation or handling of these events may be communications based, EDI processing based, or both. In addition, senders may desire to know such information on an exception basis, such as reporting only for error conditions, or they may need regular indication of the status of delivery to allow them to maintain local, internal audit information. Also, providers of communications services may need to know when interchanges for which they have accepted responsibility were forwarded and accepted by the next service provider in the transmission path, or whether forwarding was not successful.

In either scenario, the transmission or processing of interchanges can be viewed as an acknowledgment event in a general sense, creating the need for some response. From a sender’s perspective, the acceptance of their interchange by a translator or communications provider is an acknowledgment event that could either be indicated by a simple receipt, or a more thorough reporting of what actions were taken after receipt. For a service provider, forwarding interchanges can also result in an acknowledgment event being created that calls for an acknowledgment action to take place.

Taken as a set of acknowledgment requirements, these and other events can be considered as a set of circumstances which results in or require some acknowledgment action to take place. Rather than consider every possible action and event, a basic subset of these events can be defined that describes the majority of cases that form a generalized picture of tracking interchanges. Together with acknowledgment mechanisms that relate to those events and specific components that create or respond to those events, an acknowledgment model can be described.

ANSI ASC X12C has worked in this area, having produced a generic Acknowledgments Model in X12C/TG1/95-65—*Technical Report Reference Model for the Acknowledgment and Tracking of EDI Interchanges*. This technical report identifies specific entities in the EDI communications and processing path that serve as the event generators or handlers, as well as identifying X12 standards based acknowledgment mechanisms. Also, the senders and receivers of the interchanges are recognized as being the terminating application systems for which the EDI transactions are sent from or sent to, regardless of where translation occurs. The government has taken the ANSI X12 approach to an acknowledgments model, refining it through identification of specific entities and acknowledgment events. Support for this model will provide users and service providers with the ability to track interchanges and respond to requests for status of such interchanges. In addition, the internal audit trail processes of each entity will be enhanced with the availability of specified event mapping.

10.4.1 Description of Acknowledgment Model

As adapted from the generic model developed within ASC X12C, the Government Acknowledgment Model identifies specific components, acknowledgment events, and X12 mechanisms that are related to those events. Based upon the Electronic Commerce Processing Node (ECPN) as a central component, the model establishes a view of the EC/EDI Infrastructure as encompassing commercial and government entities, as well as service providers and users.

In this model, service providers are those components that provide translation services, communications services, or some EDI processing services. Specifically, the model identifies the ECPNs, VANs and Translation Points as service providers. A Service Request Handler (SRH) is a service provider whose primary function is to provide communications services between other components in the model. Users include Trading Partners (TPs) and Automated Information Systems (AISs).

The acknowledgment mechanisms identified in the model include unspecified as well as X12 based mechanisms. Where the model has identified an acknowledgment event but does not specify a mechanism for handling that event, it is implied that components involved in that event will agree on what mechanism will be used.

X12-based acknowledgment mechanisms include control segment structures in addition to transaction sets. The Interchange Delivery Notice (TA3) segment, Data Status Tracking (242) transaction set and the Functional Acknowledgment (997) transaction set all have distinct properties and functions. However, their use in a general sense as acknowledgment mechanisms allows a sequence of communications and processing events to be tied together in a logical stream. Each acknowledgment event is mapped to an X12 standards based mechanism according to where the event takes place, what type of event occurred, and what role the receiving or generating component plays in the data flow stream.

The TA3 can provide information on the status of delivery of an interchange, the time an interchange was received, or the disposition of an interchange, and is used to report such information between Service Request Handlers. The Data Status Tracking (242) transaction set, in addition to providing the ability to represent the information contained in the TA3, allows transmission status information to be conveyed from service request handlers to senders. The Functional Acknowledgment (997) transaction set indicates the status of translation of the interchange header and trailer information. These mechanisms are more fully described later in this section.

The model, as depicted in Figures 10.4-1, 10.4-2, 10.4-3, and 10.4-4, identifies the sets of events that, through implementation and use of the specified acknowledgment mechanisms, provides for the tracking of interchanges across the infrastructure.

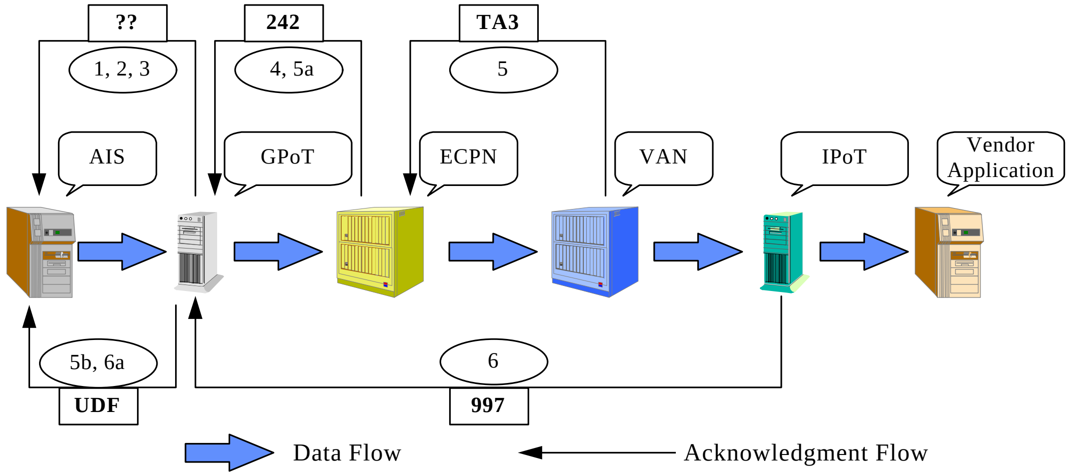
Implementation Note:

1. While the requirement for acknowledgments from GPoT *to supported AISs was identified, no single mechanism could be identified. It is therefore left to agreement between them as described in the Service Level Agreement.*

2. TA1 is not supported in this acknowledgment model implementation.

3. The government translation function can be implemented as part of the government AIS, as part of ECPN, or as a stand-alone function. GPoT acknowledgment responsibilities reside at the location performing translation.

4. The vendor translation function can be implemented as part of the Vendor Application, VAN or as a stand-alone function. IPoT acknowledgment responsibilities reside at the location performing translation.



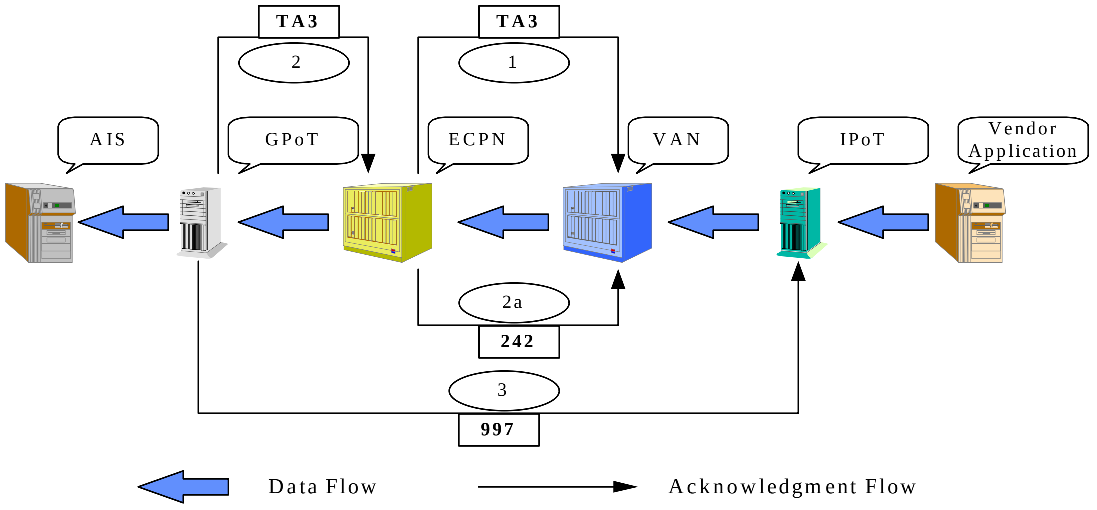
Notes:
a. The GPoT translation function may be performed by the ECPN, AIS, or by a separate entity.
b. For the purposes of the model, the govt-to-govt scenario is represented by replacing the VAN-Translation components with a GPoT component.
c. The IPoT may be operated by the VAN, the Vendor, or a third party. In all cases, the IPoT is the ultimate recipient of the interchange for the purposes of acknowledgment in this model.
d. 997s and 242s can be mapped at the GPoT to UDFs & forwarded to the AIS as agreed between the GPoT and their customer base. 242s will not be acknowledged by 997s.
e. UDF is User Defined File (flat file, proprietary file).
f. The use of 824s are not precluded by this model.
g. Support for the model acknowledgment mechanisms is mandatory. The manner of their usage is as detailed further in the Federal EDI Implementation Guidelines Part 10, or other agreements.

Figure 10.4-1 Acknowledgment Model, Commercial to Government

Sequence / Event	Mechanism	From	To
1. Receipt of UDF by GpoT	TBD	GPoT	AIS
2. Translation Result	TBD	GPoT	AIS
3. Disposition (Acknowledge that interchange has left GPoT)	TBD	GPoT	AIS
4. Interchange receipt by ECPN	242	ECPN	GPoT
5. Interchange Disposition at SRH (Government to Government)	TA3 TA3	VAN GPoT	ECPN ECPN
5a Report of Interchange Disposition at SRH	242	ECPN	GPoT
5b. Report of Interchange Disposition at SRH	UDF	GPoT	AIS
6. Translation Result	997	IPoT	GPoT
6a Translation Result	UDF	GPoT	AIS

Notes:
a. Not all events 1, 2 or 3 may occur or need to be acknowledged
b. TBD indicates the acknowledgment mechanism is to be determined, or as agreed to between components
c. UDF: User Defined File (flat file, proprietary file format)

Figure 10.4-2 Acknowledged Events, Commercial to Government



Notes:
a. Acknowledgments among VANs, Translation Points and their customers are matters to be decided by them and are not defined in the government Acknowledgment Model.
b. Some GPoTs may generate a second 242, with the ECPN acting as a pass-through.
c. For government to government scenario, replace the VAN with a GPoT. The ECPN will generate 242s in lieu of TA3s in step 1.

Figure 10.4-3 Acknowledgment Model, Government to Commercial

Sequence / Event	**Mechanism**	**From**	**To**
1. Interchange receipt by ECPN (Government to Government)	TA3 TA3	ECPN ECPN	VAN GPoT
2. Interchange Disposition at GpoT	TA3	GPoT	ECPN
2a. Report of Interchange Disposition at GPoT (Government to Government)	242 242	ECPN ECPN	VAN GPoT
3. Translation Result	997	GPoT	IPoT

Note:
In step 2a, the disposition report carried in a TA3 is mapped to a 242

Figure 10.4-4 Acknowledged Events, Government to Commercial

10.4.2 Interchange Acknowledgment

At the interchange level, acknowledgments can occur for a number of events. Successful translation, syntax error, or a more detailed acknowledgment of the disposition of an interchange can be reported. The available X12 mechanisms for such interchange acknowledgments includes the Functional Acknowledgment (997) transaction set, the Interchange Acknowledgment (TA1), and the Interchange Delivery Notice Segment (TA3). In general, the 997 is used exclusively for reporting the status of syntactical analysis of the interchange by the receiving translator, although it could be used as an indication that an interchange was received. The Interchange Acknowledgment (TA1) is not supported in this acknowledgment model. The Interchange Delivery Notice (TA3) provides the ability for reporting on the status of actions taken on a particular interchange. The manner in which these mechanisms are used, and the features within each that are utilized, provides a set of tools for building a sequence of acknowledgments for the life cycle of an interchange as it flows across an infrastructure.

Implementation Note: *The need to implement the 242 and TA3 transaction sets is under review. Federal government support for these transaction sets will be dictated by business decisions as electronic commerce continues to emerge.*

10.4.2.1 TA3

The purpose of the TA3 is to provide a notice from the receiving SRH to the sending SRH that an interchange was delivered, not delivered, refused, purged, or transferred to the next SRH. It provides a notification of action taken, notice of time/date action was taken, and the ability to report on more than one event.

As an acknowledgment mechanism in this model, the TA3 is used between the ECPN and VANs, as Service Request Handlers, to indicate the status of interchanges sent from the government to commercial components, as well as the reverse scenario. To indicate outbound delivery status, the information contained in this TA3 is further translated into a 242 transaction set and sent to GPoTs for their use, which may include supplying this information to the interchange sender. The government uses the TA3 to indicate interchange delivery status to the sending commercial infrastructure components.

Implementation Note:

1. An interchange that contains a TA3 shall contain only TA3s.

2. An interchange may contain multiple TA3s.

3. Upon delivery to the interchange receiver’s mailbox, a TA3 shall be generated.

4. If delivery to the interchange receiver’s mailbox is not made within 2 hours, a TA3 shall be generated indicating a non-delivery *status. The appropriate reason codes will be specified. A TA3 shall be generated every 2 hours indicating non-delivery status until the interchange is delivered to the receiver's mailbox. Upon delivery, Note 3 above applies.*

5. If an interchange is accepted but subsequently determined to be non-deliverable, a TA3 shall be generated indicating code RJ in TA312 and the appropriate reason code in TA303.

6. No acknowledgment is made for the receipt of a TA3.

10.4.2.2 Data Status Tracking (242) Transaction Set

The Data Status Tracking (242) transaction set conveys status information from a service request handler to the interchange sender, interchange receiver, or both. It can be used to provide status information regarding an interchange as it flows from an interchange sender through one or more service request handlers to an interchange receiver during its transmission cycle.

In the acknowledgment model, the 242 transaction set is used for two events: (1) it conveys information from the TA3 that was generated by the VAN or GPoT that received the interchange, and (2) it is used to provide acknowledgment information between government components. Because it is a transaction set, translation sites can map that information into a User Defined File (UDF) for the sending applications use. How this information is used depends on the internal business processes at the application site, and is not covered by the model. In addition, this information may be used by the GPoT in its capacity as a Service Request Handler for internal audit trail purposes.

Implementation Note*:*

1. For interchanges between government components, a 242 shall be generated upon delivery to the interchange receiver's mailbox. If delivery to the interchange receiver's mailbox is not made within 2 hours, a 242 shall be generated indicating a non-delivery status.

2. The 242 transaction set shall not be acknowledged (via a 997), nor shall it be used to report the final disposition of a 997 transaction set .

3. Additional 242 acknowledgments from interconnect service providers may be required by additional agreements among trading partners.

10.4.2.3 Interchange Acknowledgment Segment (TA1)

The Interchange Acknowledgment Segment (TA1) is used to acknowledge receipt of one interchange header and trailer envelope.

Implementation Note: *The TA1 is not supported in this acknowledgment model.*

10.4.3 Application Advice (824) Transaction Set

Although it can provide acknowledgment functionality, use of the Application Advice (824) transaction set is not specified by this model. Currently, it is primarily used on an exception basis for reporting between applications, and its full use as an acknowledgment mechanism within the model would create substantial impact on the communications and processing systems.

10.4.4 Functional Acknowledgments (997) Transaction Set

While the Functional Acknowledgment (997) transaction set is not part of the interchange control structure, it is integral to the overall process for interchange integrity, and for completeness of the acknowledgment model.

Support for the Functional Acknowledgment is required in all cases. The 997 verifies (or challenges) the syntactical correctness (e.g., ability to translate) of transaction-level data within a functional group with respect to both the implementation conventions and the ASC X12 standard.

Implementation Note*:*

1. Syntactic correctness shall be determined by comparison to the requirements of the applicable implementation convention, not simply the ASC X12 standard.

2. The 997 transaction set shall not be acknowledged.

3. When an X12 transaction containing "Not Used" segments and/or data elements and/or codes is received by the Government, the transaction will be rejected and a 997 will be generated indicating why the transaction was rejected.

4. The flow of X12 997s between sender and receiver will not be interrupted by the Government's EC infrastructure.

10.5 SECURITY

ASC X12.58, published in December, 1997, provides for the implementation of security services at the functional group and transaction set levels for X12 version 004010. The available security services include: data integrity, confidentiality, assurance, verification, and non-repudiation of origin. These services may be implemented individually or in any combination.

ASC X12.58 can meet several security objectives. Among these are:

- The recipient of an EDI transaction can verify the identity of the originator of the transaction.

- The recipient of an EDI transaction can verify the integrity of its contents.

- The originator of an EDI transaction can provide confidentiality for its contents.

ASC X12.58 provides a mechanism that can be applied to the X12 functional group or transaction set, in contrast to other alternatives which are usually applied to the entire interchange. ANSI X12.58 is transaction data independent. When X12.58 security mechanisms are applied inside the interchange, they can be handled and routed as standard X12 transactions without disrupting the end-to-end security. Since security services are applied within the interchange, they are independent of the mechanism used to transport them. Thus X12.58 can provide security even when the interchanges leave the boundaries of the ECI.

Implementation Note:

1. The federal government is committed to providing security services for ASC X12 compliant EDI via the constructs provided by ASC X12.58. However, very significant changes to those constructs have been made within various version/releases of the ASC X12 standards. Also, ASC X12.58 security constructs are not backward compatible. That is, 004010 constructs may not be applied to provide security services to the bulk of the current federal implementations, which are in version/release 003070, 003060, 003050, 003040 and earlier. Due to the evolutionary development of security structures to support ASC X12 interchanges, the Federal Government has determined that it will not support security structures prior to version/release 003070.

2. The federal government does not currently support the X12.58 security mechanism.

10.5.1 Authentication

Message authentication is a procedure to verify that received messages have not been altered. A hash function, a public function that maps a message of any length into a fixed hash value, can be used as an authenticator when used in conjunction with some form of data encryption, such as digital signature.

Implementation Note: *Assurance via the S4A/SVA segments shall be used in lieu of authentication. For 003070 implementations, assurance via the S2A/SVA shall be used.*

10.5.2 Confidentiality (Encryption)

The X12.58 standards allows for the implementation of various algorithms to encrypt X12 transactions. Cryptographic algorithms fall into two categories: secret key and public-key. Secret key algorithms are based on both the sender and receiver sharing the same secret key (i.e., key unknown to other parties). This key is used to encrypt the transaction prior to transmission and decrypt it upon receipt. Public-key algorithms are based on both sender and receiver having a pair of keys, one public and one private. All exchanges of keys between sender and receiver are limited to the public portion only, so the private key portion is protected. Encryption algorithms will be engineered to meet the requirement of compatibility with PKI. Initially, the Government will "consider" the following algorithms:

- Data Encryption Standard (DES)
- Triple DES (DE3)
- Rivest-Shamir-Adleman (RSA)
- SKIPJACK

Implementation Note:

1. Confidentiality services may be applied at either the functional group (GS/GE) level, the transaction set (ST/SE) level or both.

2. When applied, the S3S *shall be inserted immediately after the GS segment and the* S3E *shall be inserted immediately prior to the GE segment. For 003070 implementations, the S1S shall be inserted immediately after the GS segment and the S2E shall be inserted immediately prior to the GE segment.*

3. When applied, the S4S *shall be inserted immediately after the ST segment and the* S4E *shall be inserted immediately prior to the SE segment. For 003070 implementations, the S2S shall be inserted immediately after the ST segment and the S2E shall be inserted immediately prior to the SE segment.*

10.5.3 Assurance (Digital Signatures)

A digital signature is an authentication technique that also includes measures to counter repudiation by the source. Assurances as defined in X12.58, allow the originator of the transaction to express “business intent” via a digital signature.   The Government will support implementation of the Digital Signature Standard. When used, assurances are applied before any other security processes. In version 004010, the location of the group level assurance header segment (S3A) was changed. The S3A immediately follows the GS segment. The Security Value (SVA) segment follows any existing SVA segments and precedes the GE segment. This allows for efficient processing of the assurance segments. At the transaction level, the S4A segment replaces the S2A segment. The sequence of segments is detailed in Section 10.5.6

For 003070 implementations, one S2A and one SVA are inserted immediately before the SE segment of the transaction set being assured. If subsequent assurances are applied, additional S2A/SVA pairs are inserted between the previous assurance, and the SE segment of the transaction set being assured. Detailed instructions for the use of the ***assurance*** segments are contained in section 10.6

Implementation Note:

1. Assurance (digital signature) may be applied at either the functional group (GS/GE) level, the transaction set (ST/SE) level or both.

2. When both assurance and confidentiality are applied, assurance (S3A or S4A *and SVA) shall be applied first and then confidentiality (*S3S and S3E or S4S and S4E*).*

10.5.4 X12.58 Capabilities by Release

Note: The following table and notes reflect capabilities provided by the X12.58 standard. For implementation of these capabilities reference the security structures in section 10.6.

ANSI X12 Release	**Authentication**	**Encryption**	**Assurance**
003040	(Note 1)	(Note 3)	▨
003050	(Note 1)	(Note 3)	▨
003060	(Note 2)	X	X
003070	(Note 2)	X	X
004010	*(Note 2)*	*X*	*(Note 4)*

NOTES:

1. Authentication accomplished using a message authentication code (MAC). The MAC is a hash of the data.

2. Authentication accomplished as a by-product of the digital signature or by using the MAC defined in earlier releases of the ANSI X12 standard.

3. Private (symmetric) keys supported by this release. Asymmetric keying is possible but not without some “non-standard” use of data elements.

4. The assurance capability is applied via the (S3A or S4A and SVA segments.

10.5.5 Sequencing of Cryptographic Techniques

In practical situations, the users of the X12.58 standards will choose combinations of features rather than just a single feature. This is possible since all features are designed to be used in isolation or in any combination.

Authentication does not protect the confidentiality of the message because the information is interchanged in its plain text form. Message encryption can be used to provide confidentiality while using authentication to provide integrity protection of the same data. When both authentication and encryption are used, the authentication is performed before encryption of the original plain text data.

Where more than one service is selected at a specific level, the order of processing is:

a) Before applying any security services, the data must first be translated into an EDI format,
b) Addition of one or more assurances,
c) Authentication,
d) Compression,
e) Encryption, and
f) Filtering for data communications.

When assurance segments are used, they must be added to unsecured (not authenticated or encrypted) transactions. If a transaction set is received (with or without assurances) with encryption and/or authentication applied by the originator, the transaction set must be either decrypted or authenticated prior to the addition of any further assurances. Once any assurances have been added, the transaction set can be encrypted or authenticated prior to being forwarded to additional parties.

When applying security services at the functional group level, all security services at the transaction set level must be completed before applying security services at the functional group level.

The receiving organization processes the received message in the reverse order, starting with inverse filtering, followed by decryption, and then by decompression, validation of authentication and validation of the assurances. When processing inbound security services at the transaction set level, all security services at the functional group level must be removed before processing inbound security services at the transaction set level. In this manner the receiving organization unwraps the EDI message by processing the security services and removing the security segment pairs from the message before processing the next security service.

10.5.6 Transmission of Security Segments

Security services (authentication, encryption and assurances) are provided at two levels within ASC X12 in conjunction with the following envelopes:

- Functional Group (GS/GE envelope)
- Transaction Set (ST/SE envelope)

At each of these levels, authentication, encryption and assurances are each optional. Assurances are independent of authentication or encryption. In addition, any service used at one level is independent of a service used at the other level.

If encryption and/or authentication is provided, the security header segment (S3S or S4S) immediately follows the segment initiating the beginning of this level (GS or ST); the security trailer segment (S3E or S4E) precedes the segment terminating the level (GE or SE). If encryption and/or authentication at both levels is provided and if assurances are used at both levels, the sequence of segments, illustrating these levels, is:

ISA – Interchange Header

GS – Functional Group Header

S3S – Security Header Level 1

S3A – Assurance Header Level 1

ST – Transaction Set Header

S4S – Security Header Level 2

S4A – Assurance Header Level 2

(The Transaction Set Segments)

SVA – Security Value Level 2

S4E – Security Trailer Level 2

SE – Transaction Set Trailer

SVA – Security Value (Level 1)

S3E – Security Trailer Level 1

GE – Functional Group Trailer

IEA – Interchange Trailer

For 003070 implementations, if encryption and/or authentication at both levels is provided and if assurances are used at both levels, the sequence of segments, illustrating these levels, is:

ISA – Interchange Header

(Other Groups whether secured or not at Level 1)

GS – Functional Group Header

S1S – Security Header Level 1

(Other Transaction Sets whether secured or not at Level 2)

ST – Transaction Set Header

S2S – Security Header Level 2

(The Transaction Set Segments)

S2A – Security Assurance Level 2

SVA – Assurance Token Level 2

(Other optional S2A–SVA pairs at Level 2

S2E – Security Trailer Level 2

SE – Transaction Set Trailer

(Other Transaction Sets whether secured or not at Level 2)

S1A – Assurance Segment Level 1

SVA – Assurance Token Level 1

(Other optional S1A–SVA pairs at Level 1)

S1E – Security Trailer Level 1

GE – Functional Group Trailer

(Other Functional Groups whether secured or not at Level 1)

IEA – Interchange Trailer

10.6 Interchange Control, Acknowledgment and Security Segment Specifications

This section contains the implementation conventions for the:

- Interchange Control Header (ISA), Version/release 004010
- Interchange Delivery Notice Segment (TA3)
- Functional Group Header (GS), Version/release 002003 and 003010
- Functional Group Header (GS), Version/releases 003040 through 003070
- Functional Group Header (GS), Version/release 004010
- Security Header Level 1 (S1S), Version/releases 003040 and 003050
- Security Header Level 1 (S1S), Version/releases 003060 and 003070
- Security Header Level 1 (S3S), Version/release 004010
- Assurance Header Level 1(S3A), Version/release 004010
- Security Header Level 2 (S2S), Version/releases 003040 and 003050
- Security Header Level 2 (S2S), Version/releases 003060 and 003070
- Security Header Level 2 (S4S), Version/release 004010
- Security Assurance Level 2 (S2A), Version/releases 003060 and 003070
- Assurance Header Level 2 (S4A), Version/release 004010
- Assurance Token Level 2 (SVA), Version/releases 003060 and 003070
- Security Value Level 2 (SVA), Version/release 004010
- Security Trailer Level 2 (S2E), Version/releases 003040, 003050, 003060 and 003070
- Security Trailer Level 2 (S4E), Version/release 004010
- Assurance Segment Level 1 (S1A), Version/releases 003060 and 003070

- Assurance Token Level 1 (SVA), Version/releases 003060 and 003070

- Security Value Level 1 (SVA), Version/release 004010

- Security Trailer Level 1 (S1E), Version/releases 003060 and 003070

- Security Trailer Level 1 (S3E), Version/release 004010

- Functional Group Trailer (GE), Version/release 002003 through 004010

- Interchange Control Trailer (IEA), Version/release 002003 through 004010.

Segment: **ISA Interchange Control Header**

Usage: Mandatory

Max Use: 1

Purpose: To start and identify an interchange of zero or more functional groups and interchange-related control segments

Syntax Notes:

Semantic Notes:

Comments:

Notes: ***1. Use ASCII Hexadecimal 1D in the fourth byte of the Interchange Control Header. This first occurrence of an element separator dictates the value the translation software will employ throughout the interchange.***

2. Use ASCII Hexadecimal 1C after ISA16. This first occurrence of a segment terminator dictates the value the translation software employs throughout the interchange.

3. See ISA16 for subelement separator usage.

4. The ISA segment represented here is valid for version 002003-004010.

Data Element Summary

	Ref.	**Data**		
	Des.	**Element**	**Name**	**Attributes**
Must Use `	**ISA01**	**I01**	**Authorization Information Qualifier** `	**M ID 2/2**

00	No Authorization Information Present (No Meaningful Information in I02)
05	Department of Defense (DoD) Communication Identifier ***Use to indicate the Department of Defense (DOD) as the information authorizer. Use this code even if the sender is not a DOD entity.***
06	United States Federal Government Communication Identifier ***Use to indicate the Federal Government as the information authorizer. Use this code even if the sender is not a Federal Government entity.***

Must Use	**ISA02**	**I02**	**Authorization Information** Information used for additional identification or authorization of the interchange sender or the data in the interchange; the type of information is set by the Authorization Information Qualifier (I01) ***1. Use to provide additional identification or authorization for the data in the interchange. Otherwise, fill this field with blank characters.*** ***2. When used, it is recommended that the specific coding be exchanged between trading partner data security officials to ensure preservation of data security.***	**M AN 10/10**
Must Use	**ISA03**	**I03**	**Security Information Qualifier** Code to identify the type of information in the Security Information	**M ID 2/2**

00	No Security Information Present (No Meaningful

	Information in I04)
01	Password

Use based on trading partner agreement.

Must Use ISA04 I04 Security Information M AN 10/10

This is used for identifying the security information about the interchange sender or the data in the interchange; the type of information is set by the Security Information Qualifier (I03)

If ISA03 is code 00, fill this field with blank characters. Otherwise, enter a password as agreed between Trading Partners.

Must Use ISA05 I05 Interchange ID Qualifier M ID 2/2

Qualifier to designate the system/method of code structure used to designate the sender or receiver ID element being qualified

DUNS (Code 01) or DUNS+4 (Code 16) are preferred.

01	Duns (Dun & Bradstreet)
02	SCAC (Standard Carrier Alpha Code)
04	IATA (International Air Transport Association)
08	UCC EDI Communications ID (Comm ID)
09	X.121 (CCITT)
10	Department of Defense (DoD) Activity Address Code
16	Duns Number With 4-Character Suffix

Must Use ISA06 I06 Interchange Sender ID M AN 15/15

Identification code published by the sender for other parties to use as the receiver ID to route data to them; the sender always codes this value in the sender ID element

1. Enter the identifier of the sender's translation point.

2. Left justify and pad on the right with blanks.

Must Use ISA07 I05 Interchange ID Qualifier M ID 2/2

Qualifier to designate the system/method of code structure used to designate the sender or receiver ID element being qualified

DUNS (Code 01) or DUNS+4 (Code 16) are preferred.

01	Duns (Dun & Bradstreet)
02	SCAC (Standard Carrier Alpha Code)
04	IATA (International Air Transport Association)
08	UCC EDI Communications ID (Comm ID)
09	X.121 (CCITT)
10	Department of Defense (DoD) Activity Address Code
16	Duns Number With 4-Character Suffix

Must Use ISA08 I07 Interchange Receiver ID M AN 15/15

Identification code published by the receiver of the data; When sending, it is used by the sender as their sending ID, thus other parties sending to them will use this as a receiving ID to route data to them

1. Enter the identifier of the receiver's translation point (both government and non-government).

2. Left justify and pad on the right with blanks.

Must Use ISA09 I08 Interchange Date M DT 6/6

Date of the interchange

1. Express the UTC (previously known as GMT) date that this interchange was created.

2. Express the date in a six-position (YYMMDD) format.

Must Use ISA10 I09 Interchange Time M TM 4/4

Time of the interchange

1. Express the UTC (previously known as GMT) time that this interchange was created.

2. Express the time in a four-position (HHMM) format.

Must Use ISA11 I10 Interchange Control Standards Identifier M ID 1/1

Code to identify the agency responsible for the control standard used by the message that is enclosed by the interchange header and trailer

U	U.S. EDI Community of ASC X12, TDCC, and UCS

Must Use ISA12 I11 Interchange Control Version Number M ID 5/5

This version number covers the interchange control segments

Use to identify the ASC X12 version and release for the interchange envelope, not the functional group or transactions carried within the envelope.

Refer to 004010 Data Element Dictionary for acceptable code values

Must Use ISA13 I12 Interchange Control Number M N0 9/9

A control number assigned by the interchange sender

Originating activities may use any numbering scheme consistent with their business practices. However, the scheme must uniquely identify each interchange over a very long period of time.

Must Use ISA14 I13 Acknowledgment Requested M ID 1/1

Code sent by the sender to request an interchange acknowledgment (TA1)

This request for acknowledgment applies only to return of a TA1, Interchange Acknowledgment. It does not apply to other acknowledgments (e.g. TA3 or transaction set 242) as required by Part 10 of the Federal Guidelines. Since the TA1 is not supported, no acknowledgment shall be requested.

0	No Acknowledgment Requested ***Use this code to indicate an interchange acknowledgment via TA1 shall not be returned by the interchange receiver.***

Must Use ISA15 I14 Test Indicator M ID 1/1

Code to indicate whether data enclosed by this interchange envelope is test or production

P	Production Data ***Use to identify all data other than test data.***
T	Test Data ***Use when testing interchanges.***

Must Use ISA16 I15 Component Element Separator M AN 1/1

Type is not applicable; the component element separator is a delimiter and not a data element; this field provides the delimiter used to separate component data elements within a composite data structure; this value must be different than the data element separator and the segment terminator

1. ***Prior to version/release 3020 there were no component elements defined in the standard; however, the standard accommodated their future existence. It has no meaning for translation in those versions of the standard that do not have defined composite data elements. For versions that do not have composite date elements a blank character will be entered for this data element.***
2. ***Enter ASCII Hexadecimal 1F. The value of this element dictates the value the translation software employs for component element separation throughout the interchange.***

Segment: **TA3** **Interchange Delivery Notice Segment**

Usage: Mandatory

Max Use: 1

Purpose: To provide a notice from the receiving service request handler to the sending service request handler that an interchange was delivered or not delivered to the interchange receiver's mailbox, or some other ancillary service was performed, and that the interchange receiver retrieved, refused, or purged the interchange; TA3 is exchanged only between service request handlers; use of the TA3 segment is optional

Syntax Notes:
1 If either TA322 or TA323 is present, then the other is required.
2 If either TA324 or TA325 is present, then the other is required.
3 If either TA326 or TA327 is present, then the other is required.

Semantic Notes:
1 TA301 and TA302 identify the service request handlers processing the interchange being reported.
2 TA304 through TA311 and TA318 through TA321 are used to identify the interchange whose status is being reported.
3 TA312 through TA314 identify the action being reported and the date and time that action was performed. TA315 through TA317 provide a second set of interchange action code, date and time that can be included if a given TA3 is reporting on more than one event.
4 TA322 through TA327 contain optional information exchanged by service request handlers to supply additional information concerning actions taken upon the interchange being reported.

Comments:

Notes: ***1. Only one interchange action may be reported per TA3. If multiple events are to be reported, multiple TA3s must be used.***

2. Only one interchange control structure error may be reported per TA3. If multiple errors are to be reported, multiple TA3s must be used.

Data Element Summary

	Ref. Des.	Data Element	Name	Attributes
Must Use	**TA301**	**I38**	**Service Request Handler ID Qualifier** This is a code identifying the service request handler ***Cite the code FG to indicate the Federal Government. To comply with X12 syntax, do so whether the originator is a public or private organization.***	**M ID 2/2**
Must Use	**TA302**	**I39**	**Service Request Handler ID** This is the identification code of the sending service request handler ***Cite the DUNS or DUNS+4 of the service request handler providing this notice of interchange delivery.***	**M AN 1/15**
Must Use	**TA303**	**I43**	**Error Reason Code** The code indicates the error found or not found in processing the control structure or in delivery	**M ID 3/3**

000	No Errors
001	The Interchange Control Number in the Header and Trailer Do not Match; the Value from the Header is used in the Acknowledgment
002	This Standard as Noted in the Control Standards Identifier is not Supported
003	This Version of the Controls is not Supported
004	The Segment Terminator is Invalid
005	Invalid Value as Shown in the Reported Interchange Control Number

006	Invalid Value as Shown in the Reported Interchange Date
007	Invalid Value as Shown in the Reported Interchange Time
008	Invalid Value as Shown in the Reported Interchange Sender ID Qualifier
009	Invalid Value as Shown in the Reported Interchange Sender ID
010	Invalid Value as Shown in the Reported Interchange Receiver ID Qualifier
011	Invalid Value as Shown in the Reported Interchange Receiver ID
016	Trading Partnership not Established
017	Invalid Number of Included Groups Value
018	Invalid Control Structure
019	Improper (Premature) End-of-file (Transmission)
020	Duplicate Interchange Control Number
021	Invalid Data Element Separator
022	Invalid Component Element Separator
023	Failure to Transfer Interchange to the next Service Request Handler
031	Receiver Not On-line
032	Abnormal Conditions

Must Use TA304 I44 Reported Start Segment ID M AN 2/3

This contains the start segment ID of the original interchange, functional group or transaction set

For ANSI ASC X12 interchanges, the start segment ID is always ISA.

Must Use TA305 I45 Reported Control Number M AN 1/14

This is the control number value of the original interchange, functional group or transaction set

Cite the control number assigned in the original interchange control header (appearing in ISA13) for which notice is being provided. With this control number, the TA3 is linked to the original interchange envelope.

Must Use TA306 I46 Reported Date M AN 1/8

This is the date value of original interchange or functional group

Cite the date appearing in ISA09 of the interchange for which delivery notice is being provided.

Must Use TA307 I47 Reported Time M AN 1/8

This is the time value of original interchange or functional group

Cite the time appearing in ISA10 of the interchange for which delivery notice is being provided.

Must Use TA308 I48 Reported Interchange Sender ID Qualifier M AN 1/4

This is the sender ID qualifier value appearing in original interchange

Cite the value appearing in ISA05 of the interchange for which delivery notice is being provided.

Must Use TA309 I49 Reported Sender ID M AN 1/35

This is the sender ID value of original interchange or functional group

Cite the value appearing in ISA06 of the interchange for which delivery notice is being provided.

Must Use TA310 I50 Reported Interchange Receiver ID Qualifier M AN 1/4

This is the receiver ID qualifier value appearing in original interchange

Cite the value appearing in ISA07 of the interchange for which delivery

notice is being provided.

Must Use TA311 I51 Reported Receiver ID M AN 1/35

This is the receiver ID value of original interchange or functional group

Cite the value appearing in ISA08 of the interchange for which delivery notice is being provided.

Must Use TA312 I40 Action Code M ID 2/2

This is a code indicating the action taken on the interchange or functional group by the service request handler or the receiver

AK	Transfer to the Next Service Request Handler has been Acknowledged
BH	Transfer to Service Request Handler not Capable of Reporting Further Status
DL	Delivered Interchange by Service Request Handler
PU	Purged by Interchange Receiver
RD	Redirected by Service Request Handler to an Alternate Receiver as Identified in the Reference Code
RF	Refused by Interchange Receiver
RJ	Rejected by Service Request Handler; See Error Reason Code for Cause
RT	Retrieved Interchange by Receiver
TR	Transferred to Next Service Request Handler by Service Request Handler, but not yet Acknowledged

Must Use TA313 I41 Action Date M DT 6/6

This is the UTC date when the service request handler took action on the reported interchange or functional group

Express the UTC (previously known as GMT) date in a six-position (YYMMDD) format.

Must Use TA314 I42 Action Time M TM 4/6

This is the UTC time when the service request handler took action on the reported interchange or functional group

Express the UTC (previously known as GMT) time in a four-position (HHMM) format.

Not Used TA315 I40 Action Code O ID 2/2

This is a code indicating the action taken on the interchange or functional group by the service request handler or the receiver
Refer to 003070 Data Element Dictionary for acceptable code values.

Not Used TA316 I41 Action Date O DT 6/6

This is the UTC date when the service request handler took action on the reported interchange or functional group

Not Used TA317 I42 Action Time O TM 4/6

This is the UTC time when the service request handler took action on the reported interchange or functional group

Not Used TA318 I52 First Reference ID Qualifier O AN 1/4

This is the ID qualifier appearing in original interchange

Not Used TA319 I53 First Reference ID O AN 1/14

This contains information from the original interchange, as defined by the First Reference ID Qualifier data element

Not Used TA320 I54 Second Reference ID Qualifier O AN 1/4

This contains ID qualifier information appearing in original interchange

Not Used TA321 I55 Second Reference ID O AN 1/14

This contains information from the original interchange, as defined by the Second Reference ID Qualifier data element

TA322 I56 Reference Code Qualifier X ID 2/2

This is a code defining the information contained in the Reference Code data element

If TA312 is code RD, use TA322 and TA323 to identify the organization to which the interchange was redirected.

05	ID of Alternate Receiver to which Interchange Has Been Redirected

	TA323	**I57**	**Reference Code**	**X**	**AN 1/35**

This contains reference information exchanged between service request handlers concerning the reported interchange as defined by the corresponding Reference Code Qualifier data element

Cite the identifier of the organization to which the interchange was redirected. The organization shall be identified via a unique identifier from one of the sources listed as allowable codes in the ISA05 definition in section 10.6 of the Federal EDI Guidelines. The Data Universal Numbering System (DUNS) number and DUNS+4 are the preferred identifiers.

Not Used	**TA324**	**I56**	**Reference Code Qualifier**	**X**	**ID 2/2**

This is a code defining the information contained in the Reference Code data element

Not Used	**TA325**	**I57**	**Reference Code**	**X**	**AN 1/35**

This contains reference information exchanged between service request handlers concerning the reported interchange as defined by the corresponding Reference Code Qualifier data element

Not Used	**TA326**	**I56**	**Reference Code Qualifier**	**X**	**ID 2/2**

This is a code defining the information contained in the Reference Code data element

Not Used	**TA327**	**I57**	**Reference Code**	**X**	**AN 1/35**

This contains reference information exchanged between service request handlers concerning the reported interchange as defined by the corresponding Reference Code Qualifier data element

Segment:	**GS Functional Group Header**
Usage:	Mandatory
Max Use:	1
Purpose:	To indicate the beginning of a functional group and to provide control information
Syntax Notes:	
Semantic Notes:	The data interchange control number GS06 in this header must be identical to the same data element in the associated functional group trailer, GE02.
Comments:	A functional group of related transaction sets, within the scope of X12 standards, consists of a collection of similar transaction sets enclosed by a functional group header and a functional group trailer.
Notes:	***1. Use to identify the functional group containing one or more related transactions.*** ***2. Use to identify the specific implementation convention with which the transaction sets contained within the functional group envelope comply.*** ***3. The version and release of the GS segment must be the same as the version and release of the transactions that follow it as specified in the Version / Release / Industry Identifier Code (GS08).*** ***4. The GS segment represented here is valid for version 002003 and 003010.***

Data Element Summary

	Ref.	Data			
	Des.	Element	Name	Attributes	
Must Use	**GS01**	**479**	**Functional Identifier Code** Code identifying a group of application related transaction sets ***Cite any valid code defined for data element 479 within the ASC X12 standard version/release under which the transaction set is prepared. For example, the 002030 Standards Data Element Dictionary may be used for any Federal or DoD implementation convention that exists and was created based upon the 002030 standard. New codes available in the 003010 standard and not available in the 002030 standard could not be used in a 002030 based implementation convention.***	**M**	**ID 2/2**
Must Use	**GS02**	**142**	**Application Sender's Code** Code identifying party sending transmission . ***1. Cite the sending application's identifier. This identifier must be unique within the domain of the sending application's translation point. Use of a Dun and Bradstreet number (DUNS) is recommended to provide universal uniqueness.*** ***2. Transmit the required number of characters without leading or trailing blanks.***	**M**	**AN 2/12**
Must Use	**GS03**	**124**	**Application Receiver's Code** Code identifying party receiving transmission ***1. Cite the receiving application’s identifier. This identifier must be unique within the domain of the receiving application's translation point. Use of a Dun and Bradstreet number (DUNS) is recommended to provide universal uniqueness.*** ***2. Transmit the required number of characters without leading or trailing blanks.*** ***3. If the group contains PUBLIC transactions, enter the literal string***	**M**	**AN 2/12**

'PUBLIC'.

Must Use GS04 29 Data Interchange Date M DT 6/6

Date sender generated a functional group of transaction sets.

1. Enter the UTC (previously known as GMT) date that this segment was created.

2. Express the date in a six-position (YYMMDD) format.

Must Use GS05 30 Data Interchange Time M TM 4/4

Time (HHMM) when the sender generated a functional group of transaction sets (local time at sender’s location).

1. Enter the UTC (previously known as GMT) time that this segment was created.

2. Express the time in a four-position (HHMM) format.

Must Use GS06 28 Data Interchange Control Number M N0 1/9

Assigned number originated and maintained by the sender

1. Originating activities may use any numbering scheme consistent with their business practices.

2. The scheme must provide sufficient uniqueness to identify each functional group. The Group Control Number value, together with the Application Sender's and Receiver's Codes, shall be unique within an extended time frame - such as a year.

Must Use GS07 455 Responsible Agency Code M ID 1/2

Code used in conjunction with Data Element 480 to identify the issuer of the standard

X Accredited Standards Committee X12

Must Use GS08 480 Version / Release / Industry Identifier Code M AN 1/12

Code indicating the version, release, subrelease, and industry identifier of the EDI standard being used. Positions 1-3, Major Version Number; Positions 4-6, Release Level of Version; Positions 7-12, Industry or Trade Association ID (optionally assigned by user).

Each Federal and DoD Implementation Convention, based on an ANSI ASC X12 transaction set, used by the government has a unique identifier specified as follow:

Positions 1 through 6:	***ANSI ASC X12 Version and Release number (e.g. 003010) upon which the IC is based.***
Position 7:	***Organizational Scope*** ***A = APADE*** ***F = Federal*** ***D = DOD*** ***G = Government (Internal)*** ***R = Rework*** ***1 = MADES I*** ***2 = MADES II*** ***5 = SACONS 2.5*** ***6 = SACONS 2.6***
Positions 8 through 10:	***Transaction Set Identifier Code (e.g. 850).***

Position 11: ***Derivative: A character used to define the functionality of a specific implementation of a transaction set.***

If the convention is not a variant, an underscore (_) will appear in this position.

Position 12: ***A sequential number starting with 0 and incremented by 1 each time the implementation convention is revised.***

Segment: **GS** **Functional Group Header**

Usage: Mandatory

Max Use: 1

Purpose: To indicate the beginning of a functional group and to provide control information

Syntax Notes:

Semantic Notes:

1 GS04 is the group date.

2 GS05 is the group time.

3 The data interchange control number GS06 in this header must be identical to the same data element in the associated functional group trailer, GE02.

Comments:

1 A functional group of related transaction sets, within the scope of X12 standards, consists of a collection of similar transaction sets enclosed by a functional group header and a functional group trailer.

Notes:

1. Use to identify the functional group containing one or more related transactions.

2. Use to identify the specific implementation convention with which the transaction sets contained within the functional group envelope comply.

3. The version and release of the GS segment must be the same as the version and release of the transactions that follow it as specified in the Version / Release / Industry Identifier Code (GS08).

4. The GS segment represented here is valid for version 003040 through 003070.

Data Element Summary

	Ref. Des.	**Data Element**	**Name**	**Attributes**
Must Use	**GS01**	**479**	**Functional Identifier Code** Code identifying a group of application related transaction sets ***Cite any valid code defined for data element 479 within the ASC X12 standard version/release under which the transaction set is prepared. For example, the 003040 Standards Data Element Dictionary may be used for any Federal or DoD implementation convention that exists and was created based upon the 003040 standard. New codes available in the 003050 standard and not available in the 003040 standard could not be used in a 003040 based implementation convention.***	**M ID 2/2**
Must Use	**GS02**	**142**	**Application Sender's Code** Code identifying party sending transmission; codes agreed to by trading partners ***1. Cite the sending application's identifier. This identifier must be unique within the domain of the sending application's translation point. Use of a Dun and Bradstreet number (DUNS or DUNS+4) is recommended to provide universal uniqueness.*** ***2. Transmit the required number of characters without leading or trailing blanks.***	**M AN 2/15**
Must Use	**GS03**	**124**	**Application Receiver's Code** Code identifying party receiving transmission. Codes agreed to by trading partners ***1. Cite the receiving application’s identifier. This identifier must be unique within the domain of the receiving application's translation point. Use of a Dun and Bradstreet number (DUNS or DUNS+4) is recommended to provide universal uniqueness.***	**M AN 2/15**

2. Transmit the required number of characters without leading or trailing blanks.

3. If the group contains PUBLIC transactions, enter the literal string 'PUBLIC'.

Must Use   GS04   373   Date   M   DT 6/6

Date (YYMMDD)

1. Enter the UTC (previously known as GMT) date that this segment was created.

2. Express the date in a six-position (YYMMDD) format.

Must Use   GS05   337   Time   M   TM 4/8

Time expressed in 24-hour clock time as follows: HHMM, or HHMMSS, or HHMMSSD, or HHMMSSDD, where H = hours (00-23), M = minutes (00-59), S = integer seconds (00-59) and DD = decimal seconds; decimal seconds are expressed as follows: D = tenths (0-9) and DD = hundredths (00-99)

1. Enter the UTC (previously known as GMT) time that this segment was created.

2. Express the time in a four-position (HHMM) format.

Must Use   GS06   28   Group Control Number   M   N0 1/9

Assigned number originated and maintained by the sender

1. Originating activities may use any numbering scheme consistent with their business practices.

2. The scheme must provide sufficient uniqueness to identify each functional group. The Group Control Number value, together with the Application Sender's and Receiver's Codes, shall be unique within an extended time frame - such as a year.

Must Use   GS07   455   Responsible Agency Code   M   ID 1/2

Code used in conjunction with Data Element 480 to identify the issuer of the standard

X	Accredited Standards Committee X12

Must Use   GS08   480   Version / Release / Industry Identifier Code   M   AN 1/12

Code indicating the version, release, subrelease, and industry identifier of the EDI standard being used, including the GS and GE segments; if code in DE455 in GS segment is X, then in DE 480 positions 1-3 are the version number; positions 4-6 are the release and subrelease, level of the version; and positions 7-12 are the industry or trade association identifiers (optionally assigned by user); if code in DE455 in GS segment is T, then other formats are allowed

Each Federal and DoD Implementation Convention, based on an ANSI ASC X12 transaction set, used by the government has a unique identifier specified as follow:

Positions 1 through 6:   ANSI ASC X12 Version and Release number (e.g. 003010) upon which the IC is based.

Position 7:   Organizational Scope
A = APADE
F = Federal
D = DOD
G = Government (Internal)
R = Rework

1 = MADES I
2 = MADES II
5 = SACONS 2.5
6 = SACONS 2.6

Positions 8 through 10: *Transaction Set Identifier Code (e.g. 850).*

Position 11: *Derivative: A character used to define the functionality of a specific implementation of a transaction set.*

If the convention is not a variant, an underscore (_) will appear in this position.

Position 12: *A sequential number starting with 0 and incremented by 1 each time the implementation convention is revised.*

Segment: **GS** **Functional Group Header**
Usage: Mandatory
Max Use: 1
Purpose: To indicate the beginning of a functional group and to provide control information
Syntax Notes:
Semantic Notes:
1 GS04 is the group date.
2 GS05 is the group time.
3 The data interchange control number GS06 in this header must be identical to the same data element in the associated functional group trailer, GE02.

Comments:
1 A functional group of related transaction sets, within the scope of X12 standards, consists of a collection of similar transaction sets enclosed by a functional group header and a functional group trailer.

Notes:
1. Use to identify the functional group containing one or more related transactions.

2. Use to identify the specific implementation convention with which the transaction sets contained within the functional group envelope comply.

3. The version and release of the GS segment must be the same as the version and release of the transactions that follow it as specified in the Version / Release / Industry Identifier Code (GS08).

4. The GS segment represented here is valid for version 004010.

Data Element Summary

	Ref. Des.	Data Element	Name	Attributes	
Must Use	**GS01**	**479**	**Functional Identifier Code** Code identifying a group of application related transaction sets ***Cite any valid code defined for data element 479 within the ASC X12 standard version/release under which the transaction set is prepared. For example, the 004010 Standards Data Element Dictionary may be used for any Federal or DoD implementation convention that exists and was created based upon the 004010 standard.***	**M**	**ID 2/2**
Must Use	**GS02**	**142**	**Application Sender's Code** Code identifying party sending transmission; codes agreed to by trading partners ***1. Cite the sending application's identifier. This identifier must be unique within the domain of the sending application's translation point. Use of a Dun and Bradstreet number (DUNS or DUNS+4) is recommended to provide universal uniqueness.*** ***2. Transmit the required number of characters without leading or trailing blanks.***	**M**	**AN 2/15**
Must Use	**GS03**	**124**	**Application Receiver's Code** Code identifying party receiving transmission. Codes agreed to by trading partners ***1. Cite the receiving application’s identifier. This identifier must be unique within the domain of the receiving application's translation point. Use of a Dun and Bradstreet number (DUNS or DUNS+4) is recommended to provide universal uniqueness.*** ***2. Transmit the required number of characters without leading or trailing blanks.***	**M**	**AN 2/15**

3. If the group contains PUBLIC transactions, enter the literal string 'PUBLIC'.

Must Use GS04 373 Date M DT 8/8

Date (CCYYMMDD)

1. Enter the UTC (previously known as GMT) date that this segment was created.

2. Express the date in a eight-position (CCYYMMDD) format; where ‘CC’ equals the hundred-years value, ‘YY’ equals the years value, ‘MM’ equals the month value, and ‘DD’ equals the days value.

Must Use GS05 337 Time M TM 4/8

Time expressed in 24-hour clock time as follows: HHMM, or HHMMSS, or HHMMSSD, or HHMMSSDD, where H = hours (00-23), M = minutes (00-59), S = integer seconds (00-59) and DD = decimal seconds; decimal seconds are expressed as follows: D = tenths (0-9) and DD = hundredths (00-99)

1. Enter the UTC (previously known as GMT) time that this segment was created.

2. Express the time in a four-position (HHMM) format.

Must Use GS06 28 Group Control Number M N0 1/9

Assigned number originated and maintained by the sender

1. Originating activities may use any numbering scheme consistent with their business practices.

2. The scheme must provide sufficient uniqueness to identify each functional group. The Group Control Number value, together with the Application Sender's and Receiver's Codes, shall be unique within an extended time frame - such as a year.

Must Use GS07 455 Responsible Agency Code M ID 1/2

Code used in conjunction with Data Element 480 to identify the issuer of the standard

X	Accredited Standards Committee X12

Must Use GS08 480 Version / Release / Industry Identifier Code M AN 1/12

Code indicating the version, release, subrelease, and industry identifier of the EDI standard being used, including the GS and GE segments; if code in DE455 in GS segment is X, then in DE 480 positions 1-3 are the version number; positions 4-6 are the release and subrelease, level of the version; and positions 7-12 are the industry or trade association identifiers (optionally assigned by user); if code in DE455 in GS segment is T, then other formats are allowed

Each Federal and DoD Implementation Convention, based on an ANSI ASC X12 transaction set, used by the government has a unique identifier specified as follow:

Positions 1 through 6: ***ANSI ASC X12 Version and Release number (e.g. 003010) upon which the IC is based.***

Position 7: ***Organizational Scope***
A = APADE
F = Federal
D = DOD
G = Government (Internal)
R = Rework

1 = MADES I
2 = MADES II
5 = SACONS 2.5
6 = SACONS 2.6

Positions 8 through 10: ***Transaction Set Identifier Code (e.g. 850).***

Position 11: ***Derivative: A character used to define the functionality of a specific implementation of a transaction set.***

If the convention is not a variant, an underscore (_) will appear in this position.

Position 12: ***A sequential number starting with 0 and incremented by 1 each time the implementation convention is revised.***

Segment:	**S1S Security Header Level 1**
Usage:	Optional
Max Use:	1
Purpose:	To initiate the beginning of a secured area and to provide the parameters needed for authentication or encryption of a functional group
Syntax Notes:	**1** If either S1S04 or S1S05 is present, then the other is required. **2** If any of S1S06 S1S07 S1S08 or S1S09 is present, then all are required.
Semantic Notes:	**1** If S1S01 is ``AA" or ``BB", S1S04 is required. If S1S01 is ``BB" or ``EE", S1S06 is required.
Comments:	
Notes:	***1. X9 has a minimum length of 4 characters for S1S02 (the security originator); no mechanism, or registration method is provided by X9 or X12 to guarantee the uniqueness of the identifier*** ***2. X9 has a minimum length of 4 characters for S1S03 (the security recipient); no mechanism, or registration method is provided by X9 or X12 to guarantee the uniqueness of the identifier*** ***3. The S1S segment represented here is only valid for versions 003040 and 003050.***

Data Element Summary

	Ref. Des.	Data Element	Name	Attributes	
Must Use	**S1S01**	**990**	**Security Type** Code identifying the security algorithms and methods employed for this level of interchange. EE Encryption, No Authentication	**M**	**ID 2/2**
Must Use	**S1S02**	**824**	**Security Originator Name** Unique designation (identity) of the cryptographic process that performs authentication or encryption on data to be interchanged, or originates a cryptographic service message Note: X9 has a minimum length of 4 characters for the security originator; no mechanism, or registration method is provided by X9 or X12 to guarantee the uniqueness of the identifier	**M**	**AN 4/16**
Must Use	**S1S03**	**825**	**Security Recipient Name** Unique designation (identity) of the cryptographic process that performs authentication or decryption on received data, or is the destination of a cryptographic service message Note: X9 has a minimum length of 4 characters for the security recipient; no mechanism, or registration method is provided by X9 or X12 to guarantee the uniqueness of the identifier	**M**	**AN 4/16**
Not Used	**S1S04**	**991**	**Authentication Key Name** Name of the key used for authentication. This name is mutually known to the security originator and the security recipient, is unique for this relationship, and allows a particular key to be specified.	**X**	**AN 1/16**
Not Used	**S1S05**	**992**	**Authentication Service Code** Authentication option	**X**	**ID 1/1**
Must Use	**S1S06**	**993**	**Encryption Key Name** Name of the key used for encryption. This name is mutually known to the security originator and the security recipient, is unique for this relationship, and allows a particular key to be specified. ***If FORTEZZA is to be used, SKIPJACK is specified here.***	**X**	**AN 1/16**
Must Use	**S1S07**	**994**	**Encryption Service Code**	**X**	**ID 1/3**

Coded values representing options for encryption processing. The code defines the encryption mode and the transmission filter specification for filtering binary ciphertext data into transmittable text.

21	ANSI X9.23 Cipher Block Chaining (CBC), Hexadecimal Filter
22	ANSI X9.23 Cipher Block Chaining (CBC), ASCII Filter
41	ANSI X9.23 CFB-8 (Cipher Feedback), Hexadecimal Filter
42	ANSI X9.23 CFB-8 (Cipher Feedback), ASCII Filter
ZZ	Mutually Defined ***Use to specify hexadecimal filtered SKIPJACK.***

Must Use S1S08 995 Length of Data (LOD) X N 1/18

Length of data is the number of character positions of the encrypted, filtered text.

If SKIPJACK is specified in the S1S07, the length of data includes the length of the Initialization Vector (S1S09).

Must Use S1S09 996 Initialization Vector (IV) X AN 16/16

The archival representation of a 64-bit value expressed in hexadecimal notation as 16 ASCII characters from the set of characters (0..9, A..F); the 64-bit value is used as a starting point for encryption of a data sequence to increase security by introducing cryptographic variance and to synchronize cryptographic equipment; a new Initialization Vector (IV) shall be used for each message; the IV shall not be intentionally reused; the 64-bit binary value, not its ASCII representation, is used for the cryptographic process; in the interchange process, the resultant encrypted and filtered 64-bit IV is sent; the hexadecimal notation is the representation for archiving purposes; the IV shall be a random or pseudo-random number; when encrypted, the IV must be decrypted using the Electronic Code Book (ECB) mode and the same key used to encrypt the message

1. Although the current 16 byte Initialization Vector (S1S09) was designed to support DES (ANSI X9.23) encryption, the SKIPJACK algorithm can be used. SKIPJACK requires a 24 byte S1S09 field. If SKIPJACK is specified in the S1S07, sixteen ASCII 'F' characters are entered in the S1S09 field. This sets all the bits high in this field. A segment terminator follows the S1S09 field. Immediately following the segment terminator will be a 24 byte S1S09 field and the SKIPJACK encrypted data. There is no data element separator between the S1S09 and the encrypted data. An S1E (security trailer), the ASCII character 'Z' and the segment terminator follow the SKIPJACK encrypted data.

2. If a transaction is encrypted at the group level, the ST and SE will be encrypted and not accessible.

Segment: **S1S** **Security Header Level 1**

Usage: Optional

Max Use: 1

Purpose: To initiate the beginning of a secured area and to provide the parameters needed for authentication or encryption of a functional group

Syntax Notes:
1 If either S1S04 or S1S05 is present, then the other is required.
2 If any of S1S06 S1S07 S1S08 or S1S09 is present, then all are required.
3 If either C03204 or C03205 is present, then the other is required.
4 If either C03206 or C03207 is present, then the other is required.

Semantic Notes:
1 If S1S01 is "AA", "BB", "AC" or "BC", then S1S04 is required.
If S1S01 is "BB", "EE", "AC" or "EC", then S1S06 is required.

Comments:
1 X9 has a required minimum length of four characters for S1S02 (security originator). No mechanism, or registration method, is provided by X9 or X12 to guarantee uniqueness of the identifier.
2 X9 has a required minimum length of four characters for S1S03 (security recipient). No mechanism, or registration method, is provided by X9 or X12 to guarantee uniqueness of the identifier.
3 In S1S04, the special name "01234567890ABCDEF" is reserved for the hexadecimal value 01234567890ABCDEF (i.e., a fixed, nonsecret value) to provide a well-known value for data-integrity testing only.

Notes:
1. X9 has a minimum length of 4 characters for S1S02 (the security originator); no mechanism, or registration method is provided by X9 or X12 to guarantee the uniqueness of the identifier
2. X9 has a minimum length of 4 characters for S1S03 (the security recipient); no mechanism, or registration method is provided by X9 or X12 to guarantee the uniqueness of the identifier
3. The S1S segment represented here is only valid for versions 003060 and 003070.

Data Element Summary

	Ref. Des.	Data Element	Name	Attributes
Must Use	**S1S01**	**990**	**Security Type** Code identifying the security algorithms and methods applied for this level of interchange EC — No Authentication, Compression, Encryption EE — No Authentication, No Compression, Encryption	**M ID 2/2**
Must Use	**S1S02**	**824**	**Security Originator Name** Unique designation (identity) of the cryptographic process that performs authentication or encryption on data to be interchanged, or originates a cryptographic service message Note: X9 has a minimum length of 4 characters for the security originator; no mechanism, or registration method is provided by X9 or X12 to guarantee the uniqueness of the identifier	**M AN 1/64**
	S1S03	**825**	**Security Recipient Name** Unique designation (identity) of the cryptographic process that performs authentication or decryption on received data, or is the destination of a cryptographic service message Note: X9 has a minimum length of 4 characters for the security recipient; no mechanism, or registration method is provided by X9 or X12 to guarantee the uniqueness of the identifier	**O AN 1/64**
Not Used	**S1S04**	**991**	**Authentication Key Name**	**X AN 1/64**

Name of the key used for authentication; this name is mutually known to the security originator and the security recipient, is unique for this relationship, and is intended to allow the changing of the key from time to time

Note: The special key name "0123456789ABCDEF" is reserved for the hexidecimal value 0123456789ABCDEF (i.e. a fixed non-secret value) to provide a well-known value for data integrity testing only)

Not Used S1S05 992 Authentication Service Code X ID 1/1

Authentication options

Must Use S1S06 C031 Encryption Key Information X

Information needed to identify or obtain the encryption key

Must Use C03101 993 Encryption Key Name M AN 1/64

Name of the key used for encryption; this name is mutually known to the security originator and the security recipient, is unique for this relationship, and is intended to allow the changing of the key from time to time

Note: If any of the optional fields are present, the Key Name should contain either "PUBLIC" if a public key is being used to encrpyt the one-time key or the actual name of the asymmetric key-encrypting-key used to encrypt the one-time key.

If FORTEZZA is to be used, SKIPJACK is specified here.

C03102 1564 Protocol ID O ID 3/3

Code specifying protocol used to encrypt the session key

KEA	Key Encryption Algorithm
RSA	RSA Algorithm

C03103 1565 Look-up Value O AN 1/512

Value used to identify a certificate containing the public key used to encrypt the one-time key

C03104 1566 Keying Material O AN 1/512

Additional material required for decrypting the one-time key

For SKIPJACK users, the Randomized value (Ra) is placed in this data element in hexadecimal format.

C03105 1567 One-time Encryption Key O AN 1/512

Hexadecimally filtered encrypted one-time key

Must Use S1S07 C032 Encryption Service Information X

Information required by the encryption operation

Must Use C03201 994 Encryption Service Code M ID 1/3

Coded values representing options for encryption processing, including the use of compression and filtering; the code either defines the encryption mode and the transmission filter specification for filtering binary data into transmittable text or specifics that the following subelements define these values

21	ANSI X9.23 Cipher Block Chaining (CBC), Hexadecimal Filter
22	ANSI X9.23 Cipher Block Chaining (CBC), ASCII Filter
41	ANSI X9.23 CFB-8 (Cipher Feedback), Hexadecimal Filter
42	ANSI X9.23 CFB-8 (Cipher Feedback), ASCII Filter
ZZ	Mutually Defined

Use to specify hexadecimal filtered SKIPJACK.

C03202 1568 Algorithm ID O ID 3/3

Algorithm used for Encryption

DE3	Triple DEA
DES	Data Encryption Standard (Same as DEA) ***As specified in FIPS 46-2.***
SKJ	Skipjack

C03203 1569 Algorithm Mode of Operation O ID 3/3

Mode of Operation of the Encryption Algorithm

CBC	Cipher Block Chaining

C03204 1570 Filter ID Code X ID 3/3

Code specifying the type of filter used to convert data code values

ASB	ASCII-Baudot Filter
ASC	ASCII Filter
HDC	Hexadecimal Filter
UUE	Uuencoding
ZZZ	Mutually Defined ***Use to indicate Base 64.***

C03205 799 Version Identifier X AN 1/30

Revision level of a particular format, program, technique or algorithm

C03206 1571 Compression ID X ID 3/3

Type of Compression Used

913	X9E13 Compression as defined by X9.32
ZZZ	Mutually Defined ***Use to indicate that each block has been compressed by using a combination of the Lempel-Ziv LZ77 algorithm and Huffman coding, in accordance with the Internet Engineering Task Force (IETF) Request for Comments (RFC) 1951 format.***

C03207 799 Version Identifier X AN 1/30

Revision level of a particular format, program, technique or algorithm

Cite the version of the compression algorithm cited in S1S07 (C03206) above.

Must Use S1S08 995 Length of Data X N 1/18

Length of data is the number of character positions of the compressed or encrypted/filtered text; when data is plain text, this field shall be absent

If SKIPJACK is specified in the S1S07, the length of data includes the length of the Initialization Vector (S1S09).

Must Use S1S09 996 Initialization Vector X AN 16/16

The archival representation of a 64-bit value expressed in hexadecimal notation as 16 ASCII characters from the set of characters (0..9, A..F); the 64-bit value is used as a starting point for encryption of a data sequence to increase security by introducing cryptographic variance and to synchronize cryptographic equipment; a new Initialization Vector (IV) shall be used for each message; the IV shall not be intentionally reused; the 64-bit binary value, not its ASCII representation, is used for the cryptographic process; in the interchange process, the resultant encrypted and filtered 64-bit IV is sent; the hexadecimal notation is the representation for archiving purposes; the IV shall be a random or pseudo-random number; when encrypted, the IV must be decrypted using the

Electronic Code Book (ECB) mode and the same key used to encrypt the message

1. Although the current 16 byte Initialization Vector (S1S09) was designed to support DES (ANSI X9.23) encryption, the SKIPJACK algorithm can be used. SKIPJACK requires a 24 byte S1S09 field. If SKIPJACK is specified in the S1S07, sixteen ASCII 'F' characters are entered in the S1S09 field. This sets all the bits high in this field. A segment terminator follows the S1S09 field. Immediately following the segment terminator will be a 24 byte S1S09 field and the SKIPJACK encrypted data. There is no data element separator between the S1S09 and the encrypted data. An S1E (security trailer), the ASCII character 'Z' and the segment terminator follow the SKIPJACK encrypted data.

2. If a transaction is encrypted at the group level, the ST and SE will be encrypted and not accessible.

Segment: **S3S** **Security Header Level 1**

Position: 030

Loop:

Level:

Usage: Optional

Max Use: 1

Purpose: To initiate the beginning of a secured area and to provide the parameters needed for authentication or encryption of a functional group

Syntax Notes:

1 If either S3S05 or S3S06 is present, then the other is required.
2 If any of S3S08 S3S09 S3S10 or S3S11 is present, then all are required.
3 If any of C05005 C05006 C05007 or C05008 is present, then all are required.
4 If any of C05009 C05010 C05011 or C05012 is present, then all are required.
5 If either C03204 or C03205 is present, then the other is required.
6 If either C03206 or C03207 is present, then the other is required.

Semantic Notes:

1 If S3S02 is "AA", "BB", "AC", or "BC", then S3S05 is required.
If S3S02 is "BB", "EE", "AC", or "EC", then S3S08 is required.
If S3S02 is "CC" then S3S09 is required.

Comments:

1 X9 has a required minimum length of four characters for S3S03 (security originator). No mechanism, or registration method, is provided by X9 or X12 to guarantee uniqueness of the identifier.

2 X9 has a required minimum length of four characters for S3S04 (security recipient). No mechanism, or registration method, is provided by X9 or X12 to guarantee uniqueness of the identifier.

3 In S3S05, the special name "01234567890ABCDEF" is reserved for the hexadecimal value 01234567890ABCDEF (i.e., a fixed, non-secret value) to provide a well-known value for data-integrity testing only.

Notes: ***The S3S segment represented here is only valid for version 004010.***

Data Element Summary

	Ref. **Des.**	**Data** **Element**	**Name**	**Attributes**
Must Use	**S3S01**	**1621**	**Security Version/Release Identifier Code** Code indicating the version/release of the ASC X12 standard that is being used for this specific security structure. The version/release identified for this segment also applies to any corresponding trailer or security value segment. This version/release is independent of any other version/release identified in another security segment at the transaction set or functional group level. This version/release is independent of the version/release identified at the interchange or functional group level Refer to 004010 Data Element Dictionary for acceptable code values.	**M ID 6/6**
Must Use	**S3S02**	**990**	**Security Type Code** Code identifying the security algorithms and methods applied for this level of interchange EC No Authentication, Compression, Encryption EE No Authentication, No Compression, Encryption	**M ID 2/2**
Must Use	**S3S03**	**824**	**Security Originator Name** Unique designation (identity) of the cryptographic process that performs authentication or encryption on data to be interchanged, or originates a cryptographic service message Note: X9 has a minimum length of 4 characters for the security originator; no mechanism, or registration method is provided by X9 or X12 to guarantee the	**M AN 1/64**

uniqueness of the identifier

S3S04 **825** **Security Recipient Name** **O** **AN 1/64**

Unique designation (identity) of the cryptographic process that performs authentication or decryption on received data, or is the destination of a cryptographic service message

Note: X9 has a minimum length of 4 characters for the security recipient; no mechanism, or registration method is provided by X9 or X12 to guarantee the uniqueness of the identifier

Not Used **S3S05** **991** **Authentication Key Name** **X** **AN 1/64**

Name of the key used for authentication; this name is mutually known to the security originator and the security recipient, is unique for this relationship, and is intended to allow the changing of the key from time to time

Note: The special key name "0123456789ABCDEF" is reserved for the hexadecimal value 0123456789ABCDEF (i.e. a fixed non-secret value) to provide a well-known value for data integrity testing only)

Not Used **S3S06** **992** **Authentication Service Code** **X** **ID 1/1**

Authentication options

Refer to 004010 Data Element Dictionary for acceptable code values.

S3S07 **C050** **Certificate Look-up Information** **O**

Conveys the information related to or used for certificate identification

Must Use **C05001** **1675** **Look-up Value Protocol Code** **M** **ID 2/2**

Code specifying the protocol used to identify a certificate

1. It is recommended that both the AA and AC codes be used since this ensures the unique identification of the certificate owner.

2. If either AB or AC is specified, AA must also be specified. AB or AC only may not be unique across domains.

AA	X509 Issuer Distinguished Name
AB	X509 Subject Distinguished Name
AC	X509 Certificate Serial Number

Must Use **C05002** **1570** **Filter ID Code** **M** **ID 3/3**

Code specifying the type of filter used to convert data code values

HDC	Hexadecimal Filter
R64	Radix 64
ZZZ	Mutually Defined ***Used to specify no filtering.***

Must Use **C05003** **799** **Version Identifier** **M** **AN 1/30**

Revision level of a particular format, program, technique or algorithm

Must Use **C05004** **1565** **Look-up Value** **M** **AN 1/4096**

Value used to identify a certificate containing a public key

Must Use **C05005** **1675** **Look-up Value Protocol Code** **X** **ID 2/2**

Code specifying the protocol used to identify a certificate

1. It is recommended that both the AA and AC codes be used since this ensures the unique identification of the certificate owner.

2. If either AB or AC is specified, AA must also be specified. AB or AC only may not be unique across domains.

AA	X509 Issuer Distinguished Name
AB	X509 Subject Distinguished Name

			AC	X509 Certificate Serial Number		
Must Use	**C05006**	**1570**	**Filter ID Code**		**X**	**ID 3/3**
			Code specifying the type of filter used to convert data code values			
			HDC	Hexadecimal Filter		
			R64	Radix 64		
			ZZZ	Mutually Defined		
				Used to specify no filtering.		
Must Use	**C05007**	**799**	**Version Identifier**		**X**	**AN 1/30**
			Revision level of a particular format, program, technique or algorithm			
Must Use	**C05008**	**1565**	**Look-up Value**		**X**	**AN 1/4096**
			Value used to identify a certificate containing a public key			
	C05009	**1675**	**Look-up Value Protocol Code**		**X**	**ID 2/2**
			Code specifying the protocol used to identify a certificate			
			1. It is recommended that both the AA and AC codes be used since this ensures the unique identification of the certificate owner. ***2. If either AB or AC is specified, AA must also be specified. AB or AC only may not be unique across domains.***			
			AA	X509 Issuer Distinguished Name		
			AB	X509 Subject Distinguished Name		
			AC	X509 Certificate Serial Number		
	C05010	**1570**	**Filter ID Code**		**X**	**ID 3/3**
			Code specifying the type of filter used to convert data code values			
			HDC	Hexadecimal Filter		
			R64	Radix 64		
			ZZZ	Mutually Defined		
				Used to specify no filtering.		
	C05011	**799**	**Version Identifier**		**X**	**AN 1/30**
			Revision level of a particular format, program, technique or algorithm			
	C05012	**1565**	**Look-up Value**		**X**	**AN 1/4096**
			Value used to identify a certificate containing a public key			
Must Use	**S3S08**	**C031**	**Encryption Key Information**		**X**	
			To provide information needed to identify or obtain the encryption key			
Must Use	**C03101**	**993**	**Encryption Key Name**		**M**	**AN 1/64**
			Name of the key used for encryption; this name is mutually known to the security originator and the security recipient, is unique for this relationship, and is intended to allow the changing of the key from time to time Note: If any of the optional fields are present, the Key Name should contain either "PUBLIC" if a public key is being used to encrypt the one-time key or the actual name of the asymmetric key-encrypting-key used to encrypt the one-time key.			
			If FORTEZZA is to be used, SKIPJACK is specified here.			
	C03102	**1564**	**Protocol ID**		**O**	**ID 3/3**
			Code specifying protocol used to encrypt the session key			
			KEA	Key Encryption Algorithm		
			RSA	RSA Algorithm		
	C03103	**1566**	**Keying Material**		**O**	**AN 1/512**

			Additional material required for decrypting the one-time key			
			For SKIPJACK users, the Randomized value (Ra) is placed in this data element in hexadecimal format.			
	C03104	**1567**	**One-time Encryption Key**		**O**	**AN 1/512**
			Hexadecimally filtered encrypted one-time key			
Must Use	**S3S09**	**C032**	**Encryption Service Information**		**X**	
			Information required by the encryption operation			
Must Use	**C03201**	**994**	**Encryption Service Code**		**M**	**ID 1/3**
			Coded values representing options for encryption processing, including the use of compression and filtering; the code either defines the encryption mode and the transmission filter specification for filtering binary data into transmittable text or specifics that the following subelements define these values			
			21	ANSI X9.23 Cipher Block Chaining (CBC), Hexadecimal Filter		
			22	ANSI X9.23 Cipher Block Chaining (CBC), ASCII Filter		
			41	ANSI X9.23 CFB-8 (Cipher Feedback), Hexadecimal Filter		
			42	ANSI X9.23 CFB-8 (Cipher Feedback), ASCII Filter		
			ZZ	Mutually Defined ***Use to specify filtered SKIPJACK.***		
	C03202	**1568**	**Algorithm ID**		**O**	**ID 3/3**
			Algorithm used for Encryption			
			DE3	Triple DEA		
			DES	Data Encryption Standard (Same as DEA) ***As specified in FIPS 46-2.***		
			SKJ	Skipjack		
	C03203	**1569**	**Algorithm Mode of Operation**		**O**	**ID 3/3**
			Mode of Operation of the Encryption Algorithm			
			CBC	Cipher Block Chaining		
	C03204	**1570**	**Filter ID Code**		**X**	**ID 3/3**
			Code specifying the type of filter used to convert data code values			
			ASB	ASCII-Baudot Filter		
			ASC	ASCII Filter		
			HDC	Hexadecimal Filter		
			R64	Radix 64		
			UUE	UUencoding		
			ZZZ	Mutually Defined ***Use to indicate Base 64.***		
	C03205	**799**	**Version Identifier**		**X**	**AN 1/30**
			Revision level of a particular format, program, technique or algorithm			
	C03206	**1571**	**Compression ID**		**X**	**ID 3/3**
			Type of Compression Used			
			913	X9E13 Compression as defined by X9.32		
			ZZZ	Mutually Defined ***Use to indicate that each block has been compressed by using a combination of the Lempel-Ziv LZ77 algorithm and the Huffman coding, in accordance***		

		with the Internet Engineering Task Force (IETF) Request for Comments (RFC) 1951 format.		
C03207	**799**	**Version Identifier** Revision level of a particular format, program, technique or algorithm ***Cite the version of the compression algorithm cited in S3S09 (CO3206) above***	**X**	**AN 1/30**
S3S10	**995**	**Length of Data** Length of data is the number of character positions of the compressed or encrypted/filtered text; when data is plain text, this field shall be absent	**X**	**N 1/18**
S3S11	**996**	**Initialization Vector** The archival representation of a value expressed in hexadecimal notation as ASCII characters from the set of characters (0..9, A..F); the value is used as a starting point for encryption of a data sequence to increase security by introducing cryptographic variance and to synchronize cryptographic equipment; a new Initialization Vector (IV) shall be used for each message; the IV shall not be intentionally reused; the binary value, not its ASCII representation, is used for the cryptographic process; in the interchange process, the resultant encrypted and filtered IV is sent; the hexadecimal notation is the representation for archiving purposes; the IV shall be a random or pseudo-random number ***If a transaction is encrypted at the group level, the ST and SE will be encrypted and not accessible.***	**X**	**AN 16/512**

Segment: **S3A** **Assurance Header Level 1**

Position: 040

Loop:

Level:

Usage: Optional

Max Use: 1

Purpose: To allow for multiple assurances at the GSE level

Syntax Notes:

1 If any of C05005 C05006 C05007 or C05008 is present, then all are required.
2 If any of C05009 C05010 C05011 or C05012 is present, then all are required.
3 If C02804 is present, then C02803 is required.
4 If C02806 is present, then C02805 is required.
5 If C02808 is present, then C02807 is required.
6 If C02810 is present, then C02809 is required.
7 If C02812 is present, then C02811 is required.
8 If C02814 is present, then C02813 is required.
9 If C02816 is present, then C02815 is required.
10 If C02818 is present, then C02817 is required.
11 If C02820 is present, then C02819 is required.

Semantic Notes:

Comments:

1 X9 has a required minimum length of four characters for S3A05 (assurance originator). No mechanism, or registration method, is provided by X9 or X12 to guarantee uniqueness of the identifier.

2 X9 has a required minimum length of four characters for S3A06 (assurance recipient). No mechanism, or registration method, is provided by X9 or X12 to guarantee uniqueness of the identifier.

3 The date/time stamp may determine which of several key values apply, depending on start and expiration date of different key values that may share the same keyname.

4 Key distribution is performed by other means and thus only onetime keys are allowed in S3A11.

The use of particular codes and corresponding values in S3A11 is dependent on the exigencies of the various cryptographic algorithms.

Notes: ***The S3A segment represented here is only valid for version 004010.***

Data Element Summary

	Ref. Des.	Data Element	Name	Attributes
Must Use	**S3A01**	**1621**	**Security Version/Release Identifier Code** Code indicating the version/release of the ASC X12 standard that is being used for this specific security structure. The version/release identified for this segment also applies to any corresponding trailer or security value segment. This version/release is independent of any other version/release identified in another security segment at the transaction set or functional group level. This version/release is independent of the version/release identified at the interchange or functional group level Refer to 004010 Data Element Dictionary for acceptable code values.	**M ID 6/6**
Must Use	**S3A02**	**1432**	**Business Purpose of Assurance** The stated business purpose for appending the assurance to an existing secured-entity (whether functional group or transaction set); the codes represent the intention of the business or application that has control over the assurance originator	**M ID 3/3**

ASG	Authorization Signature Appropriate to this Document
CSG	Authorization Co-signature Appropriate to this Document

Must Use S3A03 C034 Computation Methods M

Algorithms used to calculate an assurance

Must Use C03401 1574 Assurance Algorithm M ID 3/3

Code specifying the algorithm used to compute the assurance token

DSS	Digital Signature Standard
	As specified in FIPS 186.
RSA	RSA

Must Use C03402 1575 Hashing Algorithm M ID 3/3

Code specifying the algorithm used to compute the assurance digest

MD5	MD5
SHA	Secure hash algorithm
	As specified in FIPS 180-1.

Must Use S3A04 1434 Domain of Computation of Assurance M ID 1/2

Code specifying the bounds of the text, whether contiguous or not, over which the computation of the Assurance Token is computed using the defined methodology of computation and any relevant Assurance Token parameters

The "body" is defined as a transaction set, beginning with the first byte of the segment immediately following the ST segment terminator and including all segments up to but not including the "S" in the first SVA segment; DO NOT include any S4A segments

The "body" can also be defined as a functional group, beginning with the first byte of the segment immediately following the GS segment terminator and including all transaction sets up to but not including the "S" in the first SVA segment at the functional group level; DO NOT include any S3A segments

"This Assurance" is defined as from the "S" in S3A or S4A up to and including the segment terminator of that segment

"Previous Assurance(s)" is defined as including the entire S3A or S4A segment and the entire corresponding SVA segment that is associated with the S3A or S4A at the same level

A	Body Only
B	Body Plus This Assurance Header Only

S3A05 1435 Assurance Originator O AN 1/64

Unique designation (identity) of the cryptographic process that performs the stated assurance on data to be interchanged

Note: X9 has a required minimum length of 4 characters for a security originator; no mechanism, or registration method, is provided by X9 or X12 to guarantee uniqueness of the identifier

S3A06 1436 Assurance Recipient O AN 1/64

Unique designation (identity) of the cryptographic process that performs validation of the stated assurance on received data. In the absence of an Assurance Recipient all potential receivers will often be able to validate the assurance because the cryptographic technique is based on a "public" (as opposed to "secret") technology

Note: X9 has required minimum length of 4 characters for a security recipient; no mechanism, or registration method, is provided by X9 or X12 to guarantee uniqueness of the identifier

	S3A07	**1443**	**Assurance Reference Number**	**O**	**AN 1/35**

Alphanumeric reference number issued by security assurance originator for the particular assurance in which it occurs; unique when used in combination with security originator data element

	S3A08	**1437**	**Date/Time Reference**	**O**	**AN 17/25**

Date/time stamp in format as follows:

YYYYMMDDHHNNSSTTTZZZ+XXXX, where YYYY = 4 digit year (with leading century), MM = month of year (01..12), DD = day of month (01..31), HH = hour of day in 24-hour format (00..23), NN = minutes of the hour (00-59), SS = second of hour (00..59), TTT = [optional] milli-seconds (000..999), ZZZ = [optional] three character, nominal timezone indicator (including daylight savings time indicator) and XXXXX = 3-5 digit (including leading + or - sign) offset of time to universal time, with three position format indicating hours-offset for whole hours, and five position format indicating hours and minutes offset where this is necessary. For example:

1993061522133OCDT+0930 which represents 15 June 1993, 22:13 (10:13pm), Central Daylight Time (Nominal Value "CDT"), in a timezone that is offset + 9:30 from Universal Time (Australia)

	S3A09	**1438**	**Assurance Text**	**O**	**AN 1/64**

Any text needed to convey the name of a signatory, registration number, certification number, or other assurance-originator defined or mutually-agreed business text related to the specific assurance; this text is not defined for X12 purposes and thus functions technically as "free form text" though it may have structure that is defined by the assurance originator, an industry group, a governmental agency, or bi-laterally between assurance originator and assurance recipient

	S3A10	**C050**	**Certificate Look-up Information**	**O**	

Conveys the information related to or used for certificate identification

Must Use	**C05001**	**1675**	**Look-up Value Protocol Code**	**M**	**ID 2/2**

Code specifying the protocol used to identify a certificate

1. It is recommended that both the AA and AC codes be used since this ensures the unique identification of the certificate owner.

2. If either AB or AC is specified, AA must also be specified. AB or AC only may not be unique across domains.

AA	X509 Issuer Distinguished Name
AB	X509 Subject Distinguished Name
AC	X509 Certificate Serial Number

Must Use	**C05002**	**1570**	**Filter ID Code**	**M**	**ID 3/3**

Code specifying the type of filter used to convert data code values

HDC	Hexadecimal Filter
R64	Radix 64
ZZZ	Mutually Defined ***Used to specify no filtering.***

Must Use	**C05003**	**799**	**Version Identifier**	**M**	**AN 1/30**

Revision level of a particular format, program, technique or algorithm

Must Use	**C05004**	**1565**	**Look-up Value**	**M**	**AN 1/4096**

Value used to identify a certificate containing a public key

Must Use	**C05005**	**1675**	**Look-up Value Protocol Code**	**X**	**ID 2/2**

			Code specifying the protocol used to identify a certificate			
			1. It is recommended that both the AA and AC codes be used since this ensures the unique identification of the certificate owner. ***2. If either AB or AC is specified, AA must also be specified. AB or AC only may not be unique across domains.***			
			AA	X509 Issuer Distinguished Name		
			AB	X509 Subject Distinguished Name		
			AC	X509 Certificate Serial Number		
Must Use	**C05006**	**1570**	**Filter ID Code**		**X**	**ID 3/3**
			Code specifying the type of filter used to convert data code values			
			HDC	Hexadecimal Filter		
			R64	Radix 64		
			ZZZ	Mutually Defined		
				Used to specify no filtering.		
Must Use	**C05007**	**799**	**Version Identifier**		**X**	**AN 1/30**
			Revision level of a particular format, program, technique or algorithm			
Must Use	**C05008**	**1565**	**Look-up Value**		**X**	**AN 1/4096**
			Value used to identify a certificate containing a public key			
	C05009	**1675**	**Look-up Value Protocol Code**		**X**	**ID 2/2**
			Code specifying the protocol used to identify a certificate			
			1. It is recommended that both the AA and AC codes be used since this ensures the unique identification of the certificate owner. ***2. If either AB or AC is specified, AA must also be specified. AB or AC only may not be unique across domains.***			
			AA	X509 Issuer Distinguished Name		
			AB	X509 Subject Distinguished Name		
			AC	X509 Certificate Serial Number		
	C05010	**1570**	**Filter ID Code**		**X**	**ID 3/3**
			Code specifying the type of filter used to convert data code values			
			HDC	Hexadecimal Filter		
			R64	Radix 64		
			ZZZ	Mutually Defined		
				Used to specify no filtering.		
	C05011	**799**	**Version Identifier**		**X**	**AN 1/30**
			Revision level of a particular format, program, technique or algorithm			
	C05012	**1565**	**Look-up Value**		**X**	**AN 1/4096**
			Value used to identify a certificate containing a public key			
	S3A11	**C028**	**Assurance Token Parameters**		**O**	
			Parameters needed to calculate the Assurance Token			
Must Use	**C02801**	**1439**	**Assurance Token Parameter Code**		**M**	**ID 2/2**
			A code specifying the type of Assurance Token Parameter			
			CI	Certification Authority ID		
			EK	Key Value - One-Time Key		
			KN	Key Name		
			NT	Notarization		
			OD	Key-Encrypting-Key for One-Time Key		

UI User ID

Must Use	**C02802**	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**M**	**AN 1/64**
	C02803	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
	C02804	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**
	C02805	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
	C02806	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**
	C02807	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
	C02808	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**
	C02809	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
	C02810	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**
	C02811	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
	C02812	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**
	C02813	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
	C02814	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**
	C02815	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
	C02816	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**
	C02817	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
	C02818	**1442**	**Assurance Token Parameter Value**	**O**	**AN 1/64**

		A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required		
C02819	**1439**	**Assurance Token Parameter Code**	**X**	**ID 2/2**
		A code specifying the type of Assurance Token Parameter		
C02820	**1442**	**Assurance Token Parameter Value**	**O**	**AN 1/64**
		A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required		

Segment: **S2S** **Security Header Level 2**

Usage: Optional

Max Use: 1

Purpose: To initiate the beginning of a secured area and to provide the parameters needed for authentication or encryption of a transaction set

Syntax Notes:
1 If either S2S04 or S2S05 is present, then the other is required.
2 If any of S2S06 S2S07 S2S08 or S2S09 is present, then all are required.

Semantic Notes:
1 If S2S01 is ``AA" or ``BB", S2S04 is required.
If S2S01 is ``BB" or ``EE", S2S06 is required.

Comments:

Notes:
1. X9 has a minimum length of 4 characters for S2S02 (the security originator); no mechanism, or registration method is provided by X9 or X12 to guarantee the uniqueness of the identifier
2. X9 has a minimum length of 4 characters for S2S03 (the security recipient); no mechanism, or registration method is provided by X9 or X12 to guarantee the uniqueness of the identifier.
3. The S2S segment represented here is only valid for versions 003040 and 003050.

Data Element Summary

	Ref.	**Data**			
	Des.	**Element**	**Name**	**Attributes**	
Must Use	**S2S01**	**990**	**Security Type** Code identifying the security algorithms and methods employed for this level of interchange. EE Encryption, No Authentication	**M**	**ID 2/2**
Must Use	**S2S02**	**824**	**Security Originator Name** Unique designation (identity) of the cryptographic process that performs authentication or encryption on data to be interchanged, or originates a cryptographic service message Note: X9 has a minimum length of 4 characters for the security originator; no mechanism, or registration method is provided by X9 or X12 to guarantee the uniqueness of the identifier	**M**	**AN 4/16**
Must Use	**S2S03**	**825**	**Security Recipient Name** Unique designation (identity) of the cryptographic process that performs authentication or decryption on received data, or is the destination of a cryptographic service message Note: X9 has a minimum length of 4 characters for the security recipient; no mechanism, or registration method is provided by X9 or X12 to guarantee the uniqueness of the identifier	**M**	**AN 4/16**
Not Used	**S2S04**	**991**	**Authentication Key Name** Name of the key used for authentication. This name is mutually known to the security originator and the security recipient, is unique for this relationship, and allows a particular key to be specified.	**X**	**AN 1/16**
Not Used	**S2S05**	**992**	**Authentication Service Code** Authentication option	**X**	**ID 1/1**
Must Use	**S2S06**	**993**	**Encryption Key Name** Name of the key used for encryption. This name is mutually known to the security originator and the security recipient, is unique for this relationship, and allows a particular key to be specified. ***If FORTEZZA is to be used, SKIPJACK is specified here.***	**X**	**AN 1/16**
Must Use	**S2S07**	**994**	**Encryption Service Code**	**X**	**ID 1/3**

Coded values representing options for encryption processing. The code defines the encryption mode and the transmission filter specification for filtering binary ciphertext data into transmittable text.

21	ANSI X9.23 Cipher Block Chaining (CBC), Hexadecimal Filter
22	ANSI X9.23 Cipher Block Chaining (CBC), ASCII Filter
41	ANSI X9.23 CFB-8 (Cipher Feedback), Hexadecimal Filter
42	ANSI X9.23 CFB-8 (Cipher Feedback), ASCII Filter
ZZ	Mutually Defined ***Use to specify hexadecimal filtered SKIPJACK.***

Must Use S2S08 995 Length of Data (LOD) X N 1/18

Length of data is the number of character positions of the encrypted, filtered text.

If SKIPJACK is specified in the S2S07, the length of data includes the length of the Initialization Vector (S2S09).

Must Use S2S09 996 Initialization Vector (IV) X AN 16/16

The archival representation of a 64-bit value expressed in hexadecimal notation as 16 ASCII characters from the set of characters (0..9, A..F); the 64-bit value is used as a starting point for encryption of a data sequence to increase security by introducing cryptographic variance and to synchronize cryptographic equipment; a new Initialization Vector (IV) shall be used for each message; the IV shall not be intentionally reused; the 64-bit binary value, not its ASCII representation, is used for the cryptographic process; in the interchange process, the resultant encrypted and filtered 64-bit IV is sent; the hexadecimal notation is the representation for archiving purposes; the IV shall be a random or pseudo-random number; when encrypted, the IV must be decrypted using the Electronic Code Book (ECB) mode and the same key used to encrypt the message

Although the current 16 byte Initialization Vector (S2S09) was designed to support DES (ANSI X9.23) encryption, the SKIPJACK algorithm can be used. SKIPJACK requires a 24 byte S2S09 field. If SKIPJACK is specified in the S2S07, sixteen ASCII 'F' characters are entered in the S2S09 field. This sets all the bits high in this field. A segment terminator follows the S2S09 field. Immediately following the segment terminator will be a 24 byte S2S09 field and the SKIPJACK encrypted data. There is no data element separator between the S2S09 and the encrypted data. An S2E (security trailer), the ASCII character 'Z' and the segment terminator follow the SKIPJACK encrypted data.

Segment: **S2S** **Security Header Level 2**

Usage: Optional

Max Use: 1

Purpose: To initiate the beginning of a secured area and to provide the parameters needed for authentication or encryption of a transaction set

Syntax Notes:
1 If either S2S04 or S2S05 is present, then the other is required.
2 If any of S2S06 S2S07 S2S08 or S2S09 is present, then all are required.
3 If either C03204 or C03205 is present, then the other is required.
4 If either C03206 or C03207 is present, then the other is required.

Semantic Notes:
1 If S2S01 is "AA", "BB", "AC" or "BC", then S2S04 is required.
If S2S01 is "BB", "EE", "AC" or "EC", then S2S06 is required.

Comments:
1 X9 has a required minimum length of four characters for S2S02 (security originator). No mechanism, or registration method, is provided by X9 or X12 to guarantee uniqueness of the identifier.
2 X9 has a required minimum length of four characters for S2S03 (security recipient). No mechanism, or registration method, is provided by X9 or X12 to guarantee uniqueness of the identifier.
3 In S2S04 the special name "01234567890ABCDEF" is reserved for the hexadecimal value 01234567890ABCDEF (i.e., a fixed nonsecret value) to provide a well-known value for data-integrity testing only.

Notes:
1. X9 has a minimum length of 4 characters for S2S02 (the security originator); no mechanism, or registration method is provided by X9 or X12 to guarantee the uniqueness of the identifier
2. X9 has a minimum length of 4 characters for S2S03 (the security recipient); no mechanism, or registration method is provided by X9 or X12 to guarantee the uniqueness of the identifier.
3. The S2S segment represented here is only valid for versions 003060 and 003070.

Data Element Summary

	Ref. Des.	**Data Element**	**Name**	**Attributes**	
Must Use	**S2S01**	**990**	**Security Type** Code identifying the security algorithms and methods applied for this level of interchange EC No Authentication, Compression, Encryption EE No Authentication, No Compression, Encryption	**M**	**ID 2/2**
Must Use	**S2S02**	**824**	**Security Originator Name** Unique designation (identity) of the cryptographic process that performs authentication or encryption on data to be interchanged, or originates a cryptographic service message Note: X9 has a minimum length of 4 characters for the security originator; no mechanism, or registration method is provided by X9 or X12 to guarantee the uniqueness of the identifier	**M**	**AN 1/64**
	S2S03	**825**	**Security Recipient Name** Unique designation (identity) of the cryptographic process that performs authentication or decryption on received data, or is the destination of a cryptographic service message Note: X9 has a minimum length of 4 characters for the security recipient; no mechanism, or registration method is provided by X9 or X12 to guarantee the uniqueness of the identifier	**O**	**AN 1/64**
Not Used	**S2S04**	**991**	**Authentication Key Name**	**X**	**AN 1/64**

			Name of the key used for authentication; this name is mutually known to the security originator and the security recipient, is unique for this relationship, and is intended to allow the changing of the key from time to time Note: The special key name "0123456789ABCDEF" is reserved for the hexidecimal value 0123456789ABCDEF (i.e. a fixed non-secret value) to provide a well-known value for data integrity testing only)		
Not Used	**S2S05**	**992**	**Authentication Service Code** Authentication options	**X**	**ID 1/1**
Must Use	**S2S06**	**C031**	**Encryption Key Information** Information needed to identify or obtain the encryption key	**X**	
Must Use	**C03101**	**993**	**Encryption Key Name** Name of the key used for encryption; this name is mutually known to the security originator and the security recipient, is unique for this relationship, and is intended to allow the changing of the key from time to time Note: If any of the optional fields are present, the Key Name should contain either "PUBLIC" if a public key is being used to encrypt the one-time key or the actual name of the asymmetric key-encrypting-key used to encrypt the one-time key. ***If FORTEZZA is to be used, SKIPJACK is specified here.***	**M**	**AN 1/64**
	C03102	**1564**	**Protocol ID** Code specifying protocol used to encrypt the session key KEA Key Encryption Algorithm RSA RSA Algorithm	**O**	**ID 3/3**
	C03103	**1565**	**Look-up Value** Value used to identify a certificate containing the public key used to encrypt the one-time key	**O**	**AN 1/512**
	C03104	**1566**	**Keying Material** Additional material required for decrypting the one-time key ***For SKIPJACK users, the Randomized value (Ra) is placed in this data element in hexadecimal format.”***	**O**	**AN 1/512**
	C03105	**1567**	**One-time Encryption Key** Hexadecimally filtered encrypted one-time key	**O**	**AN 1/512**
Must Use	**S2S07**	**C032**	**Encryption Service Information** Information required by the encryption operation	**X**	
Must Use	**C03201**	**994**	**Encryption Service Code** Coded values representing options for encryption processing, including the use of compression and filtering; the code either defines the encryption mode and the transmission filter specification for filtering binary data into transmittable text or specifies that the following subelements define these values 21 ANSI X9.23 Cipher Block Chaining (CBC), Hexadecimal Filter 22 ANSI X9.23 Cipher Block Chaining (CBC), ASCII Filter 41 ANSI X9.23 CFB-8 (Cipher Feedback), Hexadecimal Filter 42 ANSI X9.23 CFB-8 (Cipher Feedback), ASCII Filter ZZ Mutually Defined ***Use to specify hexadecimal filtered SKIPJACK.***	**M**	**ID 1/3**
	C03202	**1568**	**Algorithm ID**	**O**	**ID 3/3**

			Algorithm used for Encryption			
			DE3	Triple DEA		
			DES	Data Encryption Standard (Same as DEA)		
				As specified in FIPS 46-2.		
			SKJ	Skipjack		
	C03203	**1569**	**Algorithm Mode of Operation**		**O**	**ID 3/3**
			Mode of Operation of the Encryption Algorithm			
			CBC	Cipher Block Chaining		
	C03204	**1570**	**Filter ID Code**		**X**	**ID 3/3**
			Code specifying the type of filter used to convert data code values			
			ASB	ASCII-Baudot Filter		
			ASC	ASCII Filter		
			HDC	Hexadecimal Filter		
			UUE	Uuencoding		
			ZZZ	Mutually Defined		
				Use to indicate Base 64.		
	C03205	**799**	**Version Identifier**		**X**	**AN 1/30**
			Revision level of a particular format, program, technique or algorithm			
	C03206	**1571**	**Compression ID**		**X**	**ID 3/3**
			Type of Compression Used			
			913	X9E13 Compression as defined by X9.32		
			ZZZ	Mutually Defined		
				Use to indicate that each block has been compressed by using a combination of the Lempel-Ziv LZ77 algorithm and Huffman coding, in accordance with the Internet Engineering Task Force (IETF) Request for Comments (RFC) 1951 format.		
	C03207	**799**	**Version Identifier**		**X**	**AN 1/30**
			Revision level of a particular format, program, technique or algorithm			
			Cite the version of the compression algorithm cited in S2S07 (C03206) above.			
Must Use	**S2S08**	**995**	**Length of Data**		**X**	**N 1/18**
			Length of data is the number of character positions of the compressed or encrypted/filtered text; when data is plain text, this field shall be absent			
			If SKIPJACK is specified in the S2S07, the length of data includes the length of the Initialization Vector (S2S09).			
Must Use	**S2S09**	**996**	**Initialization Vector**		**X**	**AN 16/16**
			The archival representation of a 64-bit value expressed in hexadecimal notation as 16 ASCII characters from the set of characters (0..9, A..F); the 64-bit value is used as a starting point for encryption of a data sequence to increase security by introducing cryptographic variance and to synchronize cryptographic equipment; a new Initialization Vector (IV) shall be used for each message; the IV shall not be intentionally reused; the 64-bit binary value, not its ASCII representation, is used for the cryptographic process; in the interchange process, the resultant encrypted and filtered 64-bit IV is sent; the hexadecimal notation is the representation for archiving purposes; the IV shall be a random or pseudo-random number; when encrypted, the IV must be decrypted using the Electronic Code Book (ECB) mode and the same key used to encrypt the message			
			Although the current 16 byte Initialization Vector (S2S09) was designed to			

support DES (ANSI X9.23) encryption, the SKIPJACK algorithm can be used. SKIPJACK requires a 24 byte S2S09 field. If SKIPJACK is specified in the S2S07, sixteen ASCII 'F' characters are entered in the S2S09 field. This sets all the bits high in this field. A segment terminator follows the S2S09 field. Immediately following the segment terminator will be a 24 byte S2S09 field and the SKIPJACK encrypted data. There is no data element separator between the S2S09 and the encrypted data. An S2E (security trailer), the ASCII character 'Z' and the segment terminator follow the SKIPJACK encrypted data.

Segment: **S4S** **Security Header Level 2**
Position: 060
Loop:
Level:
Usage: Optional
Max Use: 1
Purpose: To initiate the beginning of a secured area and to provide the parameters needed for authentication or encryption of a transaction set

Syntax Notes:
1 If either S4S05 or S4S06 is present, then the other is required.
2 If any of S4S08 S4S09 S4S10 or S4S11 is present, then all are required.
3 If any of C05005 C05006 C05007 or C05008 is present, then all are required.
4 If any of C05009 C05010 C05011 or C05012 is present, then all are required.
5 If either C03204 or C03205 is present, then the other is required.
6 If either C03206 or C03207 is present, then the other is required.

Semantic Notes:
1 If S4S02 is "AA", "BB", "AC", or "BC", then S4S05 is required.
If S4S02 is "BB", "EE", "AC", or "EC", then S4S08 is required.
If S4S02 is "CC" then S4S09 is required.

Comments:
1 X9 has a required minimum length of four characters for S4S03 (security originator). No mechanism, or registration method, is provided by X9 or X12 to guarantee uniqueness of the identifier.
2 X9 has a required minimum length of four characters for S4S04 (security recipient). No mechanism, or registration method, is provided by X9 or X12 to guarantee uniqueness of the identifier.
3 In S4S05, the special name "01234567890ABCDEF" is reserved for the hexadecimal value 01234567890ABCDEF (i.e., a fixed, non-secret value) to provide a well-known value for data-integrity testing only.

Notes: ***The S4S segment represented here is only valid for version 004010.***

Data Element Summary

Ref. Des. | **Data Element** | **Name** | **Attributes**

Must Use **S4S01** **1621** **Security Version/Release Identifier Code** **M ID 6/6**
Code indicating the version/release of the ASC X12 standard that is being used for this specific security structure. The version/release identified for this segment also applies to any corresponding trailer or security value segment. This version/release is independent of any other version/release identified in another security segment at the transaction set or functional group level. This version/release is independent of the version/release identified at the interchange or functional group level

004010	Draft Standards Approved for Publication by ASC X12 Procedures Review Board through October 1997

Must Use **S4S02** **990** **Security Type Code** **M ID 2/2**
Code identifying the security algorithms and methods applied for this level of interchange

EC	No Authentication, Compression, Encryption
EE	No Authentication, No Compression, Encryption

Must Use **S4S03** **824** **Security Originator Name** **M AN 1/64**
Unique designation (identity) of the cryptographic process that performs authentication or encryption on data to be interchanged, or originates a cryptographic service message

Note: X9 has a minimum length of 4 characters for the security originator; no mechanism, or registration method is provided by X9 or X12 to guarantee the uniqueness of the identifier

S4S04 825 Security Recipient Name O AN 1/64

Unique designation (identity) of the cryptographic process that performs authentication or decryption on received data, or is the destination of a cryptographic service message

Note: X9 has a minimum length of 4 characters for the security recipient; no mechanism, or registration method is provided by X9 or X12 to guarantee the uniqueness of the identifier

Not Used S4S05 991 Authentication Key Name X AN 1/64

Name of the key used for authentication; this name is mutually known to the security originator and the security recipient, is unique for this relationship, and is intended to allow the changing of the key from time to time

Note: The special key name "0123456789ABCDEF" is reserved for the hexadecimal value 0123456789ABCDEF (i.e. a fixed non-secret value) to provide a well-known value for data integrity testing only)

Not Used S4S06 992 Authentication Service Code X ID 1/1

Authentication options

S4S07 C050 Certificate Look-up Information O

Conveys the information related to or used for certificate identification

Must Use C05001 1675 Look-up Value Protocol Code M ID 2/2

Code specifying the protocol used to identify a certificate

1. It is recommended that both the AA and AC codes be used since this ensures the unique identification of the certificate owner.

2. If either AB or AC is specified, AA must also be specified. AB or AC only may not be unique across domains.

AA	X509 Issuer Distinguished Name
AB	X509 Subject Distinguished Name
AC	X509 Certificate Serial Number
ZZ	Mutually Defined

Must Use C05002 1570 Filter ID Code M ID 3/3

Code specifying the type of filter used to convert data code values

HDC	Hexadecimal Filter
R64	Radix 64
ZZZ	Mutually Defined ***Used to specify no filtering.***

Must Use C05003 799 Version Identifier M AN 1/30

Revision level of a particular format, program, technique or algorithm

Must Use C05004 1565 Look-up Value M AN 1/4096

Value used to identify a certificate containing a public key

Must Use C05005 1675 Look-up Value Protocol Code X ID 2/2

Code specifying the protocol used to identify a certificate

1. It is recommended that both the AA and AC codes be used since this ensures the unique identification of the certificate owner.

2. If either AB or AC is specified, AA must also be specified. AB or AC only may not be unique across domains.

AA	X509 Issuer Distinguished Name
AB	X509 Subject Distinguished Name

			AC	X509 Certificate Serial Number		
Must Use	**C05006**	**1570**	**Filter ID Code** Code specifying the type of filter used to convert data code values		**X**	**ID 3/3**
			HDC	Hexadecimal Filter		
			R64	Radix 64		
			ZZZ	Mutually Defined ***Used to specify no filtering.***		
Must Use	**C05007**	**799**	**Version Identifier** Revision level of a particular format, program, technique or algorithm		**X**	**AN 1/30**
Must Use	**C05008**	**1565**	**Look-up Value** Value used to identify a certificate containing a public key		**X**	**AN 1/4096**
	C05009	**1675**	**Look-up Value Protocol Code** Code specifying the protocol used to identify a certificate ***1. It is recommended that both the AA and AC codes be used since this ensures the unique identification of the certificate owner.*** ***2. If either AB or AC is specified, AA must also be specified. AB or AC only may not be unique across domains.***		**X**	**ID 2/2**
			AA	X509 Issuer Distinguished Name		
			AB	X509 Subject Distinguished Name		
			AC	X509 Certificate Serial Number		
	C05010	**1570**	**Filter ID Code** Code specifying the type of filter used to convert data code values		**X**	**ID 3/3**
			HDC	Hexadecimal Filter		
			R64	Radix 64		
			ZZZ	Mutually Defined ***Used to specify no filtering.***		
	C05011	**799**	**Version Identifier** Revision level of a particular format, program, technique or algorithm		**X**	**AN 1/30**
	C05012	**1565**	**Look-up Value** Value used to identify a certificate containing a public key		**X**	**AN 1/4096**
Must Use	**S4S08**	**C031**	**Encryption Key Information** To provide information needed to identify or obtain the encryption key		**X**	
Must Use	**C03101**	**993**	**Encryption Key Name** Name of the key used for encryption; this name is mutually known to the security originator and the security recipient, is unique for this relationship, and is intended to allow the changing of the key from time to time Note: If any of the optional fields are present, the Key Name should contain either "PUBLIC" if a public key is being used to encrypt the one-time key or the actual name of the asymmetric key-encrypting-key used to encrypt the one-time key. ***If FORTEZZA is to be used, SKIPJACK is specified here.***		**M**	**AN 1/64**
	C03102	**1564**	**Protocol ID** Code specifying protocol used to encrypt the session key		**O**	**ID 3/3**
			KEA	Key Encryption Algorithm		
			RSA	RSA Algorithm		
	C03103	**1566**	**Keying Material**		**O**	**AN 1/512**

Additional material required for decrypting the one-time key

For SKIPJACK users, the Randomized value (Ra) is placed in this data element in hexadecimal format.

	C03104	1567	**One-time Encryption Key**	O	AN 1/512

Hexadecimally filtered encrypted one-time key

Must Use	S4S09	C032	**Encryption Service Information**	X	

Information required by the encryption operation

Must Use	C03201	994	**Encryption Service Code**	M	ID 1/3

Coded values representing options for encryption processing, including the use of compression and filtering; the code either defines the encryption mode and the transmission filter specification for filtering binary data into transmittable text or specifics that the following subelements define these values

21	ANSI X9.23 Cipher Block Chaining (CBC), Hexadecimal Filter
22	ANSI X9.23 Cipher Block Chaining (CBC), ASCII Filter
41	ANSI X9.23 CFB-8 (Cipher Feedback), Hexadecimal Filter
42	ANSI X9.23 CFB-8 (Cipher Feedback), ASCII Filter
ZZ	Mutually Defined ***Used to specify filtered SKIPJACK.***

	C03202	1568	**Algorithm ID**	O	ID 3/3

Algorithm used for Encryption

DE3	Triple DEA
DES	Data Encryption Standard (Same as DEA) ***As specified in FIPS 46-2.***
SKJ	Skipjack

	C03203	1569	**Algorithm Mode of Operation**	O	ID 3/3

Mode of Operation of the Encryption Algorithm

CBC	Cipher Block Chaining

	C03204	1570	**Filter ID Code**	X	ID 3/3

Code specifying the type of filter used to convert data code values

ASB	ASCII-Baudot Filter
ASC	ASCII Filter
HDC	Hexadecimal Filter
UUE	UUencoding
ZZZ	Mutually Defined ***Use to indicate Base 64.***

	C03205	799	**Version Identifier**	X	AN 1/30

Revision level of a particular format, program, technique or algorithm

	C03206	1571	**Compression ID**	X	ID 3/3

Type of Compression Used

Use to indicate that each block has been compressed by using a combination of the Lempel-Ziv LZ77 algorithm and the Huffman coding, in accordance with the Internet Engineering Task Force (IETF) Request for Comments (RFC) 1951 format.

Refer to 004010 Data Element Dictionary for acceptable code values.

	C03207	799	**Version Identifier**	X	AN 1/30

		Revision level of a particular format, program, technique or algorithm		
		Cite the version of the compression algorithm cited in S4S (C03206) above.		
S4S10	**995**	**Length of Data**	**X**	**N 1/18**
		Length of data is the number of character positions of the compressed or encrypted/filtered text; when data is plain text, this field shall be absent		
S4S11	**996**	**Initialization Vector**	**X**	**AN 16/512**
		The archival representation of a value expressed in hexadecimal notation as ASCII characters from the set of characters (0..9, A..F); the value is used as a starting point for encryption of a data sequence to increase security by introducing cryptographic variance and to synchronize cryptographic equipment; a new Initialization Vector (IV) shall be used for each message; the IV shall not be intentionally reused; the binary value, not its ASCII representation, is used for the cryptographic process; in the interchange process, the resultant encrypted and filtered IV is sent; the hexadecimal notation is the representation for archiving purposes; the IV shall be a random or pseudo-random number		

Segment: **S2A** **Assurance Level 2**

Usage: Optional

Max Use: 1

Purpose: To allow for multiple assurances at the ST/SE level

Syntax Notes:
1 If C02804 is present, then C02803 is required.
2 If C02806 is present, then C02805 is required.
3 If C02808 is present, then C02807 is required.
4 If C02810 is present, then C02809 is required.
5 If C02812 is present, then C02811 is required.
6 If C02814 is present, then C02813 is required.
7 If C02816 is present, then C02815 is required.
8 If C02818 is present, then C02817 is required.
9 If C02820 is present, then C02819 is required.

Semantic Notes:

Comments:
1 X9 has a required minimum length of four characters for S2A04 (security originator). No mechanism, or registration method, is provided by X9 or X12 to guarantee uniqueness of the identifier.
2 X9 has a required minimum length of four characters for S2A05 (security recipient). No mechanism, or registration method, is provided by X9 or X12 to guarantee uniqueness of the identifier.
3 The date/time stamp may determine which of several key values apply, depending on start and expiration dates of different key values that may share the same keyname.
4 Key distribution is performed by other means and thus only onetime keys are allowed in S2A09.
The use of particular codes and corresponding values in S2A09 is dependent on the exigencies of the various crytographic algorithms.

Notes: ***1. Assurance (Digital Signature) segments (S2A/SVA) are not part of the control envelope structure. When used, insert the S2A/SVA segment pair(s) immediately preceding the SE segment of the transaction set for which assurance is being provided. See Section 10.5.3 of the Federal Implementation Guidelines.***
2. The S2A segment represented here is only valid for versions 003060 and 003070.

Data Element Summary

	Ref. **Des.**	**Data** **Element**	**Name**	**Attributes**
Must Use	**S2A01**	**1432**	**Business Purpose of Assurance** The stated business purpose for appending the assurance to an existing secured-entity (whether functional group or transaction set); the codes represent the intention of the business or application that has control over the assurance originator ASG Authorization Signature Appropriate to this Document CSG Authorization Co-signature Appropriate to this Document	**M ID 3/3**
Must Use	**S2A02**	**C034**	**Computation Methods** Algorithms used to calculate an assurance	**M**
Must Use	**C03401**	**1574**	**Assurance Algorithm** Code specifying the algorithm used to compute the assurance token DSS Digital Signature Standard ***As specified in FIPS 186.*** RSA RSA	**M ID 3/3**
Must Use	**C03402**	**1575**	**Hashing Algorithm** Code specifying the algorithm used to compute the assurance digest MD5 MD5	**M ID 3/3**

SHA Secure hash algorithm

As specified in FIPS 180-1.

Must Use S2A03 1434 Domain of Computation of Assurance Digest M ID 1/2

The bounds of the text, whether contiguous or not, over which the computation of the Assurance Token is computed using the defined methodology of computation and any relevant Assurance Token parameters; the "body" is either a transaction set (beginning with the ST and including all segments up to the first S2A segment, but excluding any S2S segment) or functional group (beginning with the GS and including all transaction sets up to the first S1A segment, but excluding any S1S segment

"This Assurance" is defined as from the "S" in S1A or S2A up to and including the data element separator preceeding the assurance digest

"Previous Assurance(s)" is defined as including the entire S1A or S2A segment and the entire SVA that follows the included S1A or S2A

A Body Only

B Body plus This Assurance Only

S2A04 1435 Assurance Originator O AN 1/64

Unique designation (identity) of the cryptographic process that performs the stated assurance on data to be interchanged

Note: X9 has a required minimum length of 4 characters for a security originator; no mechanism, or registration method, is provided by X9 or X12 to guarantee uniqueness of the identifier

S2A05 1436 Assurance Recipient O AN 1/64

Unique designation (identity) of the cryptographic process that performs validation of the stated assurance on received data. In the absence of an Assurance Recipient all potential receivers will often be able to validate the assurance because the cryptographic technique is based on a "public" (as opposed to "secret") technology

Note: X9 has required minimum length of 4 characters for a security recipient; no mechanism, or registration method, is provided by X9 or X12 to guarantee uniqueness of the identifier

S2A06 1443 Assurance Reference Number O AN 1/35

Alphanumeric reference number issued by security assurance originator for the particular assurance in which it occurs; unique when used in combination with security originator data element

S2A07 1437 Date/Time Reference O AN 17/25

Date/time stamp in format as follows:

YYYYMMDDHHNNSSTTTZZZ+XXXX, where YYYY = 4 digit year (with leading century), MM = month of year (01..12), DD = day of month (01..31), HH = hour of day in 24-hour format (00..23), NN = minutes of the hour (00-59), SS = second of hour (00..59), TTT = [optional] milli-seconds (000..999), ZZZ = [optional] three character, nominal time zone indicator (including daylight savings time indicator) and XXXXX = 3-5 digit (including leading + or - sign) offset of time to universal time, with three position format indicating hours-offset for whole hours, and five position format indicating hours and minutes offset where this is necessary. For example:

			199306152213300CDT+0930 which represents 15 June 1993, 22:13 (10:13pm), Central Daylight Time (Nominal Value "CDT"), in a time zone that is offset + 9:30 from Universal Time (Australia)		
	S2A08	**1438**	**Assurance Text** Any text needed to convey the name of a signatory, registration number, certification number, or other assurance-originator defined or mutually-agreed business text related to the specific assurance; this text is not defined for X12 purposes and thus functions technically as "free form text" though it may have structure that is defined by the assurance originator, an industry group, a governmental agency, or bi-laterally between assurance originator and assurance recipient	**O**	**AN 1/64**
	S2A09	**C028**	**Assurance Token Parameters** Parameters needed to calculate the Assurance Token	**O**	
Must Use	**C02801**	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter CI Certification Authority ID EK Key Value - One-Time Key KN Key Name NT Notarization OD Key-Encrypting-Key for One-Time Key UI User ID	**M**	**ID 2/2**
Must Use	**C02802**	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**M**	**AN 1/64**
Not Used	**C02803**	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
Not Used	**C02804**	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**
Not Used	**C02805**	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
Not Used	**C02806**	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**
Not Used	**C02807**	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
Not Used	**C02808**	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**
Not Used	**C02809**	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
Not Used	**C02810**	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**
Not Used	**C02811**	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
Not Used	**C02812**	**1442**	**Assurance Token Parameter Value**	**O**	**AN 1/64**

			A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required		
Not Used	**C02813**	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
Not Used	**C02814**	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**
Not Used	**C02815**	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
Not Used	**C02816**	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**
Not Used	**C02817**	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
Not Used	**C02818**	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**
Not Used	**C02819**	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
Not Used	**C02820**	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**
	S2A10	**1440**	**Assurance Digest** The result of the application of the hash defined in the methodology expressed in ASCII-hex notation	**O**	**AN 1/512**

Segment: **S4A** **Assurance Header Level 2**
Position: 070
Loop:
Level:
Usage: Optional
Max Use: 1
Purpose: To allow for multiple assurances at the ST/SE level

Syntax Notes:
1 If any of C05005 C05006 C05007 or C05008 is present, then all are required.
2 If any of C05009 C05010 C05011 or C05012 is present, then all are required.
3 If C02804 is present, then C02803 is required.
4 If C02806 is present, then C02805 is required.
5 If C02808 is present, then C02807 is required.
6 If C02810 is present, then C02809 is required.
7 If C02812 is present, then C02811 is required.
8 If C02814 is present, then C02813 is required.
9 If C02816 is present, then C02815 is required.
10 If C02818 is present, then C02817 is required.
11 If C02820 is present, then C02819 is required.

Semantic Notes:

Comments:
1 X9 has required minimum length of four characters for S4A05 (assurance originator). No mechanism, or registration method, is provided by X9 or X12 to guarantee uniqueness of the identifier.
2 X9 has a required minimum length of four characters for S4A06 (assurance recipient). No mechanism, or registration method, is provided by X9 or X12 to guarantee uniqueness of identifier.
3 The date/time stamp may determine which of several key values apply, depending on start and expiration dates of different key values that may share the same keyname.
4 Key distribution is performed by other means and thus only onetime keys are allowed in S4A11.
The use of particular codes and corresponding values in S4A11 is dependent on the exigencies of the various cryptographic algorithms.

Notes: ***The S4A segment represented here is only valid for version 004010.***

Data Element Summary

	Ref. Des.	Data Element	Name	Attributes
Must Use	**S4A01**	**1621**	**Security Version/Release Identifier Code** Code indicating the version/release of the ASC X12 standard that is being used for this specific security structure. The version/release identified for this segment also applies to any corresponding trailer or security value segment. This version/release is independent of any other version/release identified in another security segment at the transaction set or functional group level. This version/release is independent of the version/release identified at the interchange or functional group level 004010 Draft Standards Approved for Publication by ASC X12 Procedures Review Board through October 1997	**M ID 6/6**
Must Use	**S4A02**	**1432**	**Business Purpose of Assurance** The stated business purpose for appending the assurance to an existing secured-entity (whether functional group or transaction set); the codes represent the intention of the business or application that has control over the assurance originator ASG Authorization Signature Appropriate to this Document	**M ID 3/3**

CSG Authorization Co-signature Appropriate to this Document

Must Use S4A03 C034 Computation Methods M

Algorithms used to calculate an assurance

Must Use C03401 1574 Assurance Algorithm M ID 3/3

Code specifying the algorithm used to compute the assurance token

DSS Digital Signature Standard

As specified in FIPS 186.

RSA RSA

Must Use C03402 1575 Hashing Algorithm M ID 3/3

Code specifying the algorithm used to compute the assurance digest

MD5 MD5

SHA Secure hash algorithm

As specified in FIPS 180-1.

Must Use S4A04 1434 Domain of Computation of Assurance M ID 1/2

Code specifying the bounds of the text, whether contiguous or not, over which the computation of the Assurance Token is computed using the defined methodology of computation and any relevant Assurance Token parameters

The "body" is defined as a transaction set, beginning with the first byte of the segment immediately following the ST segment terminator and including all segments up to but not including the "S" in the first SVA segment; DO NOT include any S4A segments

The "body" can also be defined as a functional group, beginning with the first byte of the segment immediately following the GS segment terminator and including all transaction sets up to but not including the "S" in the first SVA segment at the functional group level; DO NOT include any S3A segments

"This Assurance" is defined as from the "S" in S3A or S4A up to and including the segment terminator of that segment

"Previous Assurance(s)" is defined as including the entire S3A or S4A segment and the entire corresponding SVA segment that is associated with the S3A or S4A at the same level

A Body Only

B Body Plus This Assurance Header Only

S4A05 1435 Assurance Originator O AN 1/64

Unique designation (identity) of the cryptographic process that performs the stated assurance on data to be interchanged

Note: X9 has a required minimum length of 4 characters for a security originator; no mechanism, or registration method, is provided by X9 or X12 to guarantee uniqueness of the identifier

S4A06 1436 Assurance Recipient O AN 1/64

Unique designation (identity) of the cryptographic process that performs validation of the stated assurance on received data. In the absence of an Assurance Recipient all potential receivers will often be able to validate the assurance because the cryptographic technique is based on a "public" (as opposed to "secret") technology

Note: X9 has required minimum length of 4 characters for a security recipient;

no mechanism, or registration method, is provided by X9 or X12 to guarantee uniqueness of the identifier

S4A07 1443 Assurance Reference Number O AN 1/35

Alphanumeric reference number issued by security assurance originator for the particular assurance in which it occurs; unique when used in combination with security originator data element

S4A08 1437 Date/Time Reference O AN 17/25

Date/time stamp in format as follows:

YYYYMMDDHHNNSSTTTZZZ+XXXX, where YYYY = 4 digit year (with leading century), MM = month of year (01..12), DD = day of month (01..31), HH = hour of day in 24-hour format (00..23), NN = minutes of the hour (00-59), SS = second of hour (00..59), TTT = [optional] milli-seconds (000..999), ZZZ = [optional] three character, nominal timezone indicator (including daylight savings time indicator) and XXXXX = 3-5 digit (including leading + or - sign) offset of time to universal time, with three position format indicating hours-offset for whole hours, and five position format indicating hours and minutes offset where this is necessary. For example:

1993061522133OCDT+0930 which represents 15 June 1993, 22:13 (10:13pm), Central Daylight Time (Nominal Value "CDT"), in a timezone that is offset + 9:30 from Universal Time (Australia)

S4A09 1438 Assurance Text O AN 1/64

Any text needed to convey the name of a signatory, registration number, certification number, or other assurance-originator defined or mutually-agreed business text related to the specific assurance; this text is not defined for X12 purposes and thus functions technically as "free form text" though it may have structure that is defined by the assurance originator, an industry group, a governmental agency, or bi-laterally between assurance originator and assurance recipient

S4A10 C050 Certificate Look-up Information O

Conveys the information related to or used for certificate identification

Must Use C05001 1675 Look-up Value Protocol Code M ID 2/2

Code specifying the protocol used to identify a certificate

1. It is recommended that both the AA and AC codes be used since this ensures the unique identification of the certificate owner.

2. If either AB or AC is specified, AA must also be specified. AB or AC only may not be unique across domains.

AA	X509 Issuer Distinguished Name
AB	X509 Subject Distinguished Name
AC	X509 Certificate Serial Number

Must Use C05002 1570 Filter ID Code M ID 3/3

Code specifying the type of filter used to convert data code values

HDC	Hexadecimal Filter
R64	Radix 64
ZZZ	Mutually Defined ***Used to specify no filtering.***

Must Use C05003 799 Version Identifier M AN 1/30

			Revision level of a particular format, program, technique or algorithm			
Must Use	**C05004**	**1565**	**Look-up Value**		**M**	**AN 1/4096**
			Value used to identify a certificate containing a public key			
Must Use	**C05005**	**1675**	**Look-up Value Protocol Code**		**X**	**ID 2/2**
			Code specifying the protocol used to identify a certificate			
			1. It is recommended that both the AA and AC codes be used since this ensures the unique identification of the certificate owner. ***2. If either AB or AC is specified, AA must also be specified. AB or AC only may not be unique across domains.***			
			AA	X509 Issuer Distinguished Name		
			AB	X509 Subject Distinguished Name		
			AC	X509 Certificate Serial Number		
Must Use	**C05006**	**1570**	**Filter ID Code**		**X**	**ID 3/3**
			Code specifying the type of filter used to convert data code values			
			HDC	Hexadecimal Filter		
			R64	Radix 64		
			ZZZ	Mutually Defined		
				Used to specify no filtering.		
Must Use	**C05007**	**799**	**Version Identifier**		**X**	**AN 1/30**
			Revision level of a particular format, program, technique or algorithm			
Must Use	**C05008**	**1565**	**Look-up Value**		**X**	**AN 1/4096**
			Value used to identify a certificate containing a public key			
	C05009	**1675**	**Look-up Value Protocol Code**		**X**	**ID 2/2**
			Code specifying the protocol used to identify a certificate			
			1. It is recommended that both the AA and AC codes be used since this ensures the unique identification of the certificate owner. ***2. If either AB or AC is specified, AA must also be specified. AB or AC only may not be unique across domains.***			
			AA	X509 Issuer Distinguished Name		
			AB	X509 Subject Distinguished Name		
			AC	X509 Certificate Serial Number		
	C05010	**1570**	**Filter ID Code**		**X**	**ID 3/3**
			Code specifying the type of filter used to convert data code values			
			HDC	Hexadecimal Filter		
			R64	Radix 64		
			ZZZ	Mutually Defined		
				Used to specify no filtering.		
	C05011	**799**	**Version Identifier**		**X**	**AN 1/30**
			Revision level of a particular format, program, technique or algorithm			
	C05012	**1565**	**Look-up Value**		**X**	**AN 1/4096**
			Value used to identify a certificate containing a public key			
	S4A11	**C028**	**Assurance Token Parameters**		**O**	
			Parameters needed to calculate the Assurance Token			
Must Use	**C02801**	**1439**	**Assurance Token Parameter Code**		**M**	**ID 2/2**
			A code specifying the type of Assurance Token Parameter			
			CI	Certification Authority ID		

EK	Key Value - One-Time Key
KN	Key Name
NT	Notarization
OD	Key-Encrypting-Key for One-Time Key
UI	User ID

Must Use	**C02802**	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**M**	**AN 1/64**
Not Used	**C02803**	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
Not Used	**C02804**	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**
Not Used	**C02805**	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
Not Used	**C02806**	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**
Not Used	**C02807**	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
Not Used	**C02808**	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**
Not Used	**C02809**	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
Not Used	**C02810**	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**
Not Used	**C02811**	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
Not Used	**C02812**	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**
Not Used	**C02813**	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
Not Used	**C02814**	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**
Not Used	**C02815**	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
Not Used	**C02816**	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**
Not Used	**C02817**	**1439**	**Assurance Token Parameter Code**	**X**	**ID 2/2**

			A code specifying the type of Assurance Token Parameter		
Not Used	**C02818**	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**
Not Used	**C02819**	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
Not Used	**C02820**	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**

Segment:	**SVA Security Value**
Usage:	Optional
Max Use:	1
Purpose:	To provide the encoded output of a cryptographic algorithm
Syntax Notes:	
Semantic Notes:	
Comments:	
Notes:	***1. Assurance (Digital Signature) segments (S2A/SVA) are not part of the control envelope structure. When used, insert the S2A/SVA segment pair(s) immediately preceding the SE segment of the transaction set for which assurance is being provided. See Section 10.5.3 of the Federal Implementation Guidelines.*** ***2. The SVA segment represented here is only valid for versions 003060 and 003070.***

Data Element Summary

	Ref.	Data			Attributes	
	Des.	Element	Name			
Must Use	**SVA01**	**1570**	**Filter ID Code**		**M**	**ID 3/3**
			Code specifying the type of filter used to convert data code values			
			ASB	ASCII-Baudot Filter		
			ASC	ASCII Filter		
			HDC	Hexadecimal Filter		
			UUE	Uuencoding		
			ZZZ	Mutually Defined		
				Use to indicate Base 64.		
Must Use	**SVA02**	**799**	**Version Identifier**		**M**	**AN 1/30**
			Revision level of a particular format, program, technique or algorithm			
Must Use	**SVA03**	**C033**	**Security Value**		**M**	
			Value of the Security Token			
Must Use	**C03301**	**1572**	**Security Value Qualifier**		**M**	**ID 3/3**
			Type of Security Value			
			ASV	Assurance Token		
			CRT	Certificate		
			PUB	Public Key		
Must Use	**C03302**	**1573**	**Encoded Security Value**		**M**	**AN 1/1E+16**
			Encoded representation of the Security Value specified by the Security Value Qualifier			

Segment:	**SVA** **Security Value**
Position:	080
Loop:	
Level:	
Usage:	Optional
Max Use:	1
Purpose:	To provide the encoded output of a cryptographic algorithm
Syntax Notes:	
Semantic Notes:	
Comments:	
Notes:	***The SVA segment represented here is only valid for version 004010.***

Data Element Summary

	Ref. Des.	**Data Element**	**Name**		**Attributes**	
Must Use	**SVA01**	**1570**	**Filter ID Code** Code specifying the type of filter used to convert data code values		**M**	**ID 3/3**
			ASB	ASCII-Baudot Filter		
			ASC	ASCII Filter		
			HDC	Hexadecimal Filter		
			UUE	UUencoding		
			ZZZ	Mutually Defined ***Use to indicate Base 64.***		
Must Use	**SVA02**	**799**	**Version Identifier** Revision level of a particular format, program, technique or algorithm		**M**	**AN 1/30**
Must Use	**SVA03**	**C033**	**Security Value** Value of the Security Token		**M**	
Must Use	**C03301**	**1572**	**Security Value Qualifier** Type of Security Value		**M**	**ID 3/3**
			ASV	Assurance Token		
			CRT	Certificate		
			PUB	Public Key		
Must Use	**C03302**	**1573**	**Encoded Security Value** Encoded representation of the Security Value specified by the Security Value Qualifier		**M**	**AN 1/1E+16**

Segment:	**S2E Security Trailer Level 2**
Usage:	Optional
Max Use:	1
Purpose:	To end a secured area and to provide the value of cryptographically computed authentication codes
Syntax Notes:	
Semantic Notes:	
Comments:	
Notes:	***The S2E segment represented here is valid for versions 003040, 003050, 003060 and 003070.***

Data Element Summary

	Ref. **Des.**	**Data** **Element**	**Name**	**Attributes**
Must Use	**S2E01**	**997**	**Hash or Authentication Code** The message authentication code or hash/digest generated by the authentication process; when the Data Encryption Standard (DES) algorithm is used, the field consists of 4 hexadecimal coded characters (i.e., characters from the set 0..9, A..F), a separator character (space, "-", or other), and 4 hexadecimally coded characters; when non-DES hashes are used, the result of the hash is expressed as hexadecimally coded characters without spaces; when authentication or hash is not used, this field should be filled with a non-blank character other than the set (0..9, A..F) for the minimum length ***Enter the character "Z".***	**M AN 1/64**

Segment: **S4E** **Security Trailer Level 2**
Position: 090
Loop:
Level:
Usage: Optional
Max Use: 1
Purpose: To end a secured area and to provide the value of cryptographically computed authentication codes
Syntax Notes:
Semantic Notes:
Comments:
Notes: ***The S4E segment represented here is only valid for version 004010.***

Data Element Summary

	Ref. **Des.**	**Data** **Element**	**Name**	**Attributes**
Must Use	**S4E01**	**997**	**Hash or Authentication Code** The message authentication code or hash/digest generated by the authentication process; when the Data Encryption Standard (DES) algorithm is used, the field consists of 4 hexadecimal coded characters (i.e., characters from the set 0..9, A..F), a separator character (space, "-", or other), and 4 hexadecimally coded characters; when non-DES hashes are used, the result of the hash is expressed as hexadecimally coded characters without spaces; when authentication or hash is not used, this field should be filled with a non-blank character other than the set (0..9, A..F) for the minimum length ***Enter the character "Z".***	**M AN 1/64**

Segment: **S1A** **Assurance Level 1**
Usage: Optional
Max Use: 1
Purpose: To allow for multiple assurances at the GS/GE level
Syntax Notes:
1 If C02804 is present, then C02803 is required.
2 If C02806 is present, then C02805 is required.
3 If C02808 is present, then C02807 is required.
4 If C02810 is present, then C02809 is required.
5 If C02812 is present, then C02811 is required.
6 If C02814 is present, then C02813 is required.
7 If C02816 is present, then C02815 is required.
8 If C02818 is present, then C02817 is required.
9 If C02820 is present, then C02819 is required.
Semantic Notes:
Comments:
1 X9 has a required minimum length of four characters for S1A04 (security originator). No mechanism, or registration method, is provided by X9 or X12 to guarantee uniqueness of the identifier.
2 X9 has a required minimum length of four characters for S1A05 (security recipient). No mechanism, or registration method, is provided by X9 or X12 to guarantee uniqueness of the identifier.
3 The date/time stamp may determine which of several key values apply, depending on start and expiration dates of different key values that may share the same keyname.
4 Key distribution is performed by other means and thus only onetime keys are allowed in S1A09.
The use of particular codes and corresponding values in S1A09 is dependent on the exigencies of the various cryptographic algorithms.

Notes: ***1. Assurance (Digital Signature) segments (S1A/SVA) are not part of the control envelope structure. When used, insert the S1A/SVA segment pair(s) immediately preceding the GE segment of the group for which assurance is being provided. See Section 10.5.3 of the Federal Implementation Guidelines.***
2. The S1A segment represented here is only valid for versions 003060 and 003070.

Data Element Summary

	Ref. **Des.**	**Data** **Element**	**Name**		**Attributes**
Must Use	**S1A01**	**1432**	**Business Purpose of Assurance**		**M ID 3/3**
			The stated business purpose for appending the assurance to an existing secured-entity (whether functional group or transaction set); the codes represent the intention of the business or application that has control over the assurance originator		
			ASG	Authorization Signature Appropriate to this Document	
			CSG	Authorization Co-signature Appropriate to this Document	
Must Use	**S1A02**	**C034**	**Computation Methods**		**M**
			Algorithms used to calculate an assurance		
Must Use	**C03401**	**1574**	**Assurance Algorithm**		**M ID 3/3**
			Code specifying the algorithm used to compute the assurance token		
			DSS	Digital Signature Standard ***As specified in FIPS 186.***	
			RSA	RSA	
Must Use	**C03402**	**1575**	**Hashing Algorithm**		**M ID 3/3**
			Code specifying the algorithm used to compute the assurance digest		
			MD5	MD5	

SHA Secure hash algorithm

As specified in FIPS 180-1.

Must Use S1A03 1434 Domain of Computation of Assurance Digest M ID 1/2

The bounds of the text, whether contiguous or not, over which the computation of the Assurance Token is computed using the defined methodology of computation and any relevant Assurance Token parameters; the "body" is either a transaction set (beginning with the ST and including all segments up to the first S2A segment, but excluding any S2S segment) or functional group (beginning with the GS and including all transaction sets up to the first S1A segment, but excluding any S1S segment

"This Assurance" is defined as from the "S" in S1A or S2A up to and including the data element separator preceeding the assurance digest

"Previous Assurance(s)" is defined as including the entire S1A or S2A segment and the entire SVA that follows the included S1A or S2A

A Body Only

B Body plus This Assurance Only

S1A04 1435 Assurance Originator O AN 1/64

Unique designation (identity) of the cryptographic process that performs the stated assurance on data to be interchanged

Note: X9 has a required minimum length of 4 characters for a security originator; no mechanism, or registration method, is provided by X9 or X12 to guarantee uniqueness of the identifier

S1A05 1436 Assurance Recipient O AN 1/64

Unique designation (identity) of the cryptographic process that performs validation of the stated assurance on received data. In the absence of an Assurance Recipient all potential receivers will often be able to validate the assurance because the cryptographic technique is based on a "public" (as opposed to "secret") technology

Note: X9 has required minimum length of 4 characters for a security recipient; no mechanism, or registration method, is provided by X9 or X12 to guarantee uniqueness of the identifier

S1A06 1443 Assurance Reference Number O AN 1/35

Alphanumeric reference number issued by security assurance originator for the particular assurance in which it occurs; unique when used in combination with security originator data element

S1A07 1437 Date/Time Reference O AN 17/25

Date/time stamp in format as follows:

YYYYMMDDHHNNSSTTTZZZ+XXXX, where YYYY = 4 digit year (with leading century), MM = month of year (01..12), DD = day of month (01..31), HH = hour of day in 24-hour format (00..23), NN = minutes of the hour (00-59), SS = second of hour (00..59), TTT = [optional] milli-seconds (000..999), ZZZ = [optional] three character, nominal time zone indicator (including daylight savings time indicator) and XXXXX = 3-5 digit (including leading + or - sign) offset of time to universal time, with three position format indicating hours-offset for whole hours, and five position format indicating hours and minutes offset where this is necessary. For example:

199306152213300CDT+0930 which represents 15 June 1993, 22:13 (10:13pm), Central Daylight Time (Nominal Value "CDT"), in a timezone that is offset + 9:30 from Universal Time (Australia)

S1A08 **1438** **Assurance Text** **O** **AN 1/64**

Any text needed to convey the name of a signatory, registration number, certification number, or other assurance-originator defined or mutually-agreed business text related to the specific assurance; this text is not defined for X12 purposes and thus functions technically as "free form text" though it may have structure that is defined by the assurance originator, an industry group, a governmental agency, or bi-laterally between assurance originator and assurance recipient

S1A09 **C028** **Assurance Token Parameters** **O**

Parameters needed to calculate the Assurance Token

Must Use **C02801** **1439** **Assurance Token Parameter Code** **M** **ID 2/2**

A code specifying the type of Assurance Token Parameter

CI	Certification Authority ID
EK	Key Value - One-Time Key
KN	Key Name
NT	Notarization
OD	Key-Encrypting-Key for One-Time Key
UI	User ID

Must Use **C02802** **1442** **Assurance Token Parameter Value** **M** **AN 1/64**

A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required

Not Used **C02803** **1439** **Assurance Token Parameter Code** **X** **ID 2/2**

A code specifying the type of Assurance Token Parameter

Not Used **C02804** **1442** **Assurance Token Parameter Value** **O** **AN 1/64**

A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required

Not Used **C02805** **1439** **Assurance Token Parameter Code** **X** **ID 2/2**

A code specifying the type of Assurance Token Parameter

Not Used **C02806** **1442** **Assurance Token Parameter Value** **O** **AN 1/64**

A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required

Not Used **C02807** **1439** **Assurance Token Parameter Code** **X** **ID 2/2**

A code specifying the type of Assurance Token Parameter

Not Used **C02808** **1442** **Assurance Token Parameter Value** **O** **AN 1/64**

A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required

Not Used **C02809** **1439** **Assurance Token Parameter Code** **X** **ID 2/2**

A code specifying the type of Assurance Token Parameter

Not Used **C02810** **1442** **Assurance Token Parameter Value** **O** **AN 1/64**

A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required

Not Used **C02811** **1439** **Assurance Token Parameter Code** **X** **ID 2/2**

A code specifying the type of Assurance Token Parameter

Not Used **C02812** **1442** **Assurance Token Parameter Value** **O** **AN 1/64**

			A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required		
Not Used	**C02813**	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
Not Used	**C02814**	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**
Not Used	**C02815**	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
Not Used	**C02816**	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**
Not Used	**C02817**	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
Not Used	**C02818**	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**
Not Used	**C02819**	**1439**	**Assurance Token Parameter Code** A code specifying the type of Assurance Token Parameter	**X**	**ID 2/2**
Not Used	**C02820**	**1442**	**Assurance Token Parameter Value** A value of a parameter, usually specifying one or more options, required for the proper operation of the cryptographic algorithm used to compute the Assurance Token; depending on the algorithm used, one or more values may be required	**O**	**AN 1/64**
	S1A10	**1440**	**Assurance Digest** The result of the application of the hash defined in the methodology expressed in ASCII-hex notation	**O**	**AN 1/512**

Segment:	**SVA Security Value**
Usage:	Optional
Max Use:	1
Purpose:	To provide the encoded output of a cryptographic algorithm
Syntax Notes:	
Semantic Notes:	
Comments:	
Notes:	***1. Assurance (Digital Signature) segments (S1A/SVA) are not part of the control envelope structure. When used, insert the S1A/SVA segment pair(s) immediately preceding the GE segment of the transaction set for which assurance is being provided. See Section 10.5.3 of the Federal Implementation Guidelines.*** ***2. The SVA segment represented here is valid for versions 003060 and 003070.***

Data Element Summary

	Ref. Des.	**Data Element**	**Name**		**Attributes**	
Must Use	**SVA01**	**1570**	**Filter ID Code**		**M**	**ID 3/3**
			Code specifying the type of filter used to convert data code values			
			ASB	ASCII-Baudot Filter		
			ASC	ASCII Filter		
			HDC	Hexadecimal Filter		
			UUE	Uuencoding		
			ZZZ	Mutually Defined		
				Use to indicate Base 64.		
Must Use	**SVA02**	**799**	**Version Identifier**		**M**	**AN 1/30**
			Revision level of a particular format, program, technique or algorithm			
Must Use	**SVA03**	**C033**	**Security Value**		**M**	
			Value of the Security Token			
Must Use	**C03301**	**1572**	**Security Value Qualifier**		**M**	**ID 3/3**
			Type of Security Value			
			ASV	Assurance Token		
			CRT	Certificate		
				Only for use in the 003070 version of this segment.		
			PUB	Public Key		
				Only for use in the 003070 version of this segment.		
Must Use	**C03302**	**1573**	**Encoded Security Value**		**M**	**AN 1/1E+16**
			Encoded representation of the Security Value specified by the Security Value Qualifier			

Segment:	**SVA** **Security Value**
Position:	080
Loop:	
Level:	
Usage:	Optional
Max Use:	1
Purpose:	To provide the encoded output of a cryptographic algorithm
Syntax Notes:	
Semantic Notes:	
Comments:	
Notes:	***The SVA segment represented here is valid for version 004010.***

Data Element Summary

	Ref.	**Data**				
	Des.	**Element**	**Name**		**Attributes**	
Must Use	**SVA01**	**1570**	**Filter ID Code**		**M**	**ID 3/3**
			Code specifying the type of filter used to convert data code values			
			ASB	ASCII-Baudot Filter		
			ASC	ASCII Filter		
			HDC	Hexadecimal Filter		
			UUE	UUencoding		
			ZZZ	Mutually Defined		
				Use to indicate Base 64.		
Must Use	**SVA02**	**799**	**Version Identifier**		**M**	**AN 1/30**
			Revision level of a particular format, program, technique or algorithm			
Must Use	**SVA03**	**C033**	**Security Value**		**M**	
			Value of the Security Token			
Must Use	**C03301**	**1572**	**Security Value Qualifier**		**M**	**ID 3/3**
			Type of Security Value			
			ASV	Assurance Token		
			CRT	Certificate		
			PUB	Public Key		
Must Use	**C03302**	**1573**	**Encoded Security Value**		**M**	**AN 1/1E+16**
			Encoded representation of the Security Value specified by the Security Value Qualifier			

Segment: **S1E** **Security Trailer Level 1**
Usage: Optional
Max Use: 1
Purpose: To end a secured area and to provide the value of cryptographically computed authentication codes
Syntax Notes:
Semantic Notes:
Comments:
Notes: ***The S1E segment represented here is valid for versions 003060 and 003070.***

Data Element Summary

	Ref.	**Data**			
	Des.	**Element**	**Name**	**Attributes**	
Must Use	**S1E01**	**997**	**Hash or Authentication Code** The message authentication code or hash/digest generated by the authentication process; when the Data Encryption Standard (DES) algorithm is used, the field consists of 4 hexadecimal coded characters (i.e., characters from the set 0..9, A..F), a separator character (space, "-", or other), and 4 hexadecimally coded characters; when non-DES hashes are used, the result of the hash is expressed as hexadecimally coded characters without spaces; when authentication or hash is not used, this field should be filled with a non-blank character other than the set (0..9, A..F) for the minimum length ***Enter the character "Z".***	**M**	**AN 1/64**

Segment:	**S3E** **Security Trailer Level 1**
Position:	120
Loop:	
Level:	
Usage:	Optional
Max Use:	1
Purpose:	To end a secured area and to provide the value of cryptographically computed authentication codes
Syntax Notes:	
Semantic Notes:	
Comments:	
Notes:	***The S3E segment represented here is only valid for version 004010.***

Data Element Summary

	Ref. **Des.**	**Data** **Element**	**Name**	**Attributes**
Must Use	**S3E01**	**997**	**Hash or Authentication Code** The message authentication code or hash/digest generated by the authentication process; when the Data Encryption Standard (DES) algorithm is used, the field consists of 4 hexadecimal coded characters (i.e., characters from the set 0..9, A..F), a separator character (space, "-", or other), and 4 hexadecimally coded characters; when non-DES hashes are used, the result of the hash is expressed as hexadecimally coded characters without spaces; when authentication or hash is not used, this field should be filled with a non-blank character other than the set (0..9, A..F) for the minimum length ***Enter the character "Z".***	**M AN 1/64**

Segment: **GE Functional Group Trailer**

Usage: Mandatory

Max Use: 1

Purpose: To indicate the end of a functional group and to provide control information

Syntax Notes:

Semantic Notes: **1** The data interchange control number GE02 in this trailer must be identical to the same data element in the associated functional group header, GS06.

Comments: **1** The use of identical data interchange control numbers in the associated functional group header and trailer is designed to maximize functional group integrity. The control number is the same as that used in the corresponding header.

Notes: ***The GE segment represented here is valid for version 002003 through 004010.***

Data Element Summary

	Ref.	Data		
	Des.	Element	Name	Attributes
Must Use	**GE01**	**97**	**Number of Transaction Sets Included** Total number of transaction sets included in the functional group or interchange (transmission) group terminated by the trailer containing this data element ***1. Use to identify the number of ST segments (transactions) within a functional group.*** ***2. Transmit the required number of characters without leading or trailing blanks.***	**M N0 1/6**
Must Use	**GE02**	**28**	**Group Control Number** Assigned number originated and maintained by the sender ***Cite the same group control number as was assigned by the originator in GS06.***	**M N0 1/9**

Segment: **IEA Interchange Control Trailer**

Usage: Mandatory

Max Use: 1

Purpose: To define the end of an interchange of zero or more functional groups and interchange-related control segments

Syntax Notes:

Semantic Notes:

Comments:

Notes: ***The IEA segment represented here is valid for version 002003 through 004010.***

Data Element Summary

	Ref. Des.	Data Element	Name	Attributes
Must Use	**IEA01**	**I16**	**Number of Included Functional Groups** A count of the number of functional groups included in an interchange ***1. Use to identify the number of GS segments (functional groups) within an interchange.*** ***2. Transmit the required number of characters without leading or trailing blanks.***	**M N0 1/5**
Must Use	**IEA02**	**I12**	**Interchange Control Number** A control number assigned by the interchange sender ***Cite the same nine-digit interchange control number as was assigned by the originator in ISA13.***	**M N0 9/9**